MINISTRY OF EDUCATION AND TRAINING

FPT UNIVERSITY

Capstone Project Document

Family Tree

Group Members	Trần Đức Long	01752	Teamlead
	Nguyễn Phương Dung	01603	Member
	Nguyễn Thế Trung	01246	Member
	Phạm Đình Đức	01559	Member
	Trần Thị Thu Huyền	60510	Member
Supervisor	Trần Đình Trí		
Ext Supervisor			
Capstone Project code	FT2013		

Contents

Contents **2**

Records of changes **7**

Chapter 1: Introduction **8**

1.1. Project information 8

1.2. Purpose 8

1.3. Members 8

1.4. Background 9

1.5. Literature Review of Existing Systems 10

1.6. Proposal and Expected system 10

1.7. Scope 11

Chapter 2: Software Project Management Plan **12**

2.1. Problem Definition 12

2.1.1. Name of this Capstone Project 12

2.1.2. Project Abstract 12

2.1.2.1. Problem definition 12

2.1.2.2. User and their problem 12

2.1.2.2.1. Family user 12

2.1.2.2.2. Family administrator 13

2.1.2.3. What is the goal? 13

2.1.2.4. What are the side-effects to be avoided? 13

2.1.2.5. Which actions are admissible? 13

2.1.3. Project Overview 13

2.1.3.1. The current system: 13

2.1.3.2. The Proposed System 14

2.1.3.3. Boundaries of the System 14

2.1.3.4. Development Environment 14

2.1.3.4.1. Hardware requirement: 14

2.1.3.4.2. Software requirement: 14

2.1.3.4.3. Testing requirement: 15

2.2. Project Organization 15

2.2.1. Software Process Model 15

2.2.2. Roles and Responsibilities 16
2.2.3. Tools and Techniques 16
2.3. Project Management Plan 16
2.3.1. Development Phases and Input/output 16
2.3.1.1 Initiating 16
2.3.1.2 Planning 16
2.3.1.3 User Specifications Requirement 17
2.3.1.4 Software Requirements Specifications 17
2.3.1.5 Design Database 17
2.3.1.6 GUI design 17
2.3.1.7 Coding 17
2.3.1.8 Testing and Fix bugs 18
2.3.1.9 Software User's Manual 18
2.3.1.10 Deployment 18
2.3.1.11 Submit work 19
2.3.2. TaskSheet 20
2.3.3. All Meeting Minutes 22
Chapter 3: Software Requirements Specifications 23
3.1. User Requirement Specification 23
3.1.1 Overview 23
3.1.2 Functional Requirements 23
3.2. System Requirement Specification 24
3.3. Software System Attributes 68
3.3.1. Usability 68
3.3.2. Maintainability 68
3.3.3. Security 68
3.3.4. Portability 68
3.3.5. Performance 68
3.4. Entity Relationship Diagram 69
Chapter 4: Software Design Description (SDD) 70
4.1. Design Overview 70
4.2. System Architectural Design 70
4.2.1. System Architectural **72**

4.2.1.1. 3 Layer Windows Application in C#.NET 74
4.2.2. Alternative Designs 75
4.2.3. System Interface 75
4.3. Component Diagrams 76
4.4. Detailed Description of Components 77
4.4.1. CRC Cards (Class-Responsibility-Collaborators) 77
4.4.1.1. Component 1: Authentication 80
4.4.1.1.1. Class Diagram 80
4.4.3.1.2. Flow-Chart 80
4.4.1.2. Component 2: Person Management 82
4.4.1.2.1. Class Diagram 82
4.4.1.2.2. Flow-Chart 82
4.4.1.3. Component-3: Tree Management 86
4.4.1.3.1. Class Diagram 86
4.4.1.3.2. Class Diagram Explanation 86
4.4.1.3.3. Flow-Chart 86
4.4.1.4. Component-4: Interface Management 90
4.4.1.4.1. Class Diagram 90
4.4.1.4.2. Class Diagram Explanation 90
4.4.1.4.3. Flow-Chart 90
4.4.1.5. Component-5: Account Management 92
4.4.1.5.1. Class Diagram 92
4.4.1.5.2. Class Diagram Explanation 92
4.4.1.5.3. Flow-Chart 92
4.4.1.6. Component-6: Database Access 94
4.4.1.6.1. Class Diagram 94
4.4.1.6.2. Flow-Chart 95
4.5. Sequence Diagram 96
4.5.1. For use case "*Login*" 96
4.5.2. For use case "*Create Node*" 96
4.5.3. For use case "*Log out*" 97
4.5.4. For use case "*Create new user account*" 97

4.5.5. For use case "*Edit user account*" **98**
4.5.6. For use case "*Delete user account*" **98**
4.5.7. For use case "*Edit Node*" **99**
4.5.8. For use case "*View Tree*" **99**
4.5.9. For use case "*Delete Tree*" **100**
4.5.10. For use case "*Send Comment*" **100**
4.5.11. For use case "*Review Comment*" **101**
4.5.12. For use case "*Search*" **101**
4.6. User Interface Design/Hardware Interface Design **101**
4.6.1. Description of the User Interface **101**
4. 6.1.1. Screen Image/Hardware Description **102**
4.6.1.1.1. "Home Page" screen **102**
4.6.1.1.2. "About" screen **103**
4.6.1.1.3. "Manage User" screen **103**
4.6.1.1.4. "Manage Tree" screen **104**
4.6.1.1.5. Notifications **104**
4.7. Database Design or Data Structure or algorithms **105**
4.7.1. Database Design **105**
4.7.2. Algorithms **106**
4.7.2.1. Algorithm 1: DFS (Depth-first search) **106**
4.7.2.2. Algorithm 2: Recursive tree construction algorithms **106**
4.7.2.2.1. CraftTree function **106**
4.7.2.2.3. Load Children-Partner function **107**
Chapter 5: Software Test Document……110
5.Introduction **108**
5.1.1.System Overview **108**
5.1.2.Test Approach **108**
5.2.Test plan **110**
5.2.1.Feature to be tested **110**
5.2.1.1.Items **110**
5.2.1.1.1.System Administrator functions **110**
5.2.1.1.2.Family Administrator functions **110**

5.2.1.2.Family User function 110
5.2.2.Feature not to be tested 111
5.2.3.Testing Tool and Environment 111
5.2.3.1.Testing tool 111
5.2.3.2.Environment 111
5.2.3.3.Test strategy 111
5.2.3.4.Test types 111
5.2.3.5.Test stages: 111
5.2.3.6.Risk list 112
5.2.3.7.Testing document management 112
5.2.3.8.Resource and responsibilities 112
5.2.3.9.Test milestones 112
5.2.3.10.Deliverables 113
5.3.Test cases 113
5.3.1.Purpose 113
5.3.2.Inputs 113
5.3.3.Expected Outputs & Pass/Fail criteria 113
5.3.4.Test Procedure 114
5.4.Checklists 114
5.4.1.Checklist of Validation 114
5.4.2.Submission Checklist 114
5.4.3.Other material 115
5.4.3.1.Log for test 115
5.4.3.2.Test Results 115
5.4.3.2.1Unit test case 116
5.4.3.2.2.Integration and System test case 119
5.4.3.2.3. Incident Report 124
Chapter 6: Software User's Manual 125

Records of changes

*A - Added M - Modified D – Deleted

Date	Change Item	A,M,D	Description	By	Version
2013-05-13		A	Create new document (report1)	DungNP	0.1
2013-05-18	2	A	Adding project plan (report 2)	HuyenTTT	0.2
2013-06-02	2.1	M	Edit problem definition	HuyenTTT	
2013-06-05	3.	A	Adding Software requirement specifications (report 3)	DungNP	0.3
2013-06-22	3.	M	Update use cases: Search, Approve family administrator	DungNP	0.4
2013-06-25	4.	A	Adding System Design Description (Report 4)	DungNP	0.5
2013-07-02	4.	M	Edit component diagram, sequence diagram	DungNP	0.6
2013-07-30	5.	A	Adding Software Test Document	HuyenTTT	0.7

Chapter 1: Introduction

1.1. Project information

- Project name: Family Tree
- Project code: FT2013
- Project type: Website application
- Timeline: May 2013~ August 2013

1.2. Purpose

This project is registered and implemented as the capstone project for the team members. The first purpose is fulfilling the requirements of FPT University studying program. The second purpose is creating a completed product for going live.

1.3. Members

	Full name	Phone	E-mail	Role in Group
Supervisor	Trần Đình Trí	0913.0919.52	Tridt@fpt.edu.vn	Supervisor
Student 1	Trần Đức Long	01646.903.539	LongTD01752@fpt.edu.vn	Teamlead
Student 2	Nguyễn Phương Dung	0169.7525.259	DungNP01603@fpt.edu.vn	Member
Student 3	Nguyễn Thế Trung	0915.127.614	TrungNT01246@fpt.edu.vn	Member
Student 4	Phạm Đình Đức	094.585.2626	DucPD01559@fpt.edu.vn	Member
Student 5	Trần Thị Thu Huyền	0983.886.949	HuyenTTT60510@fpt.edu.vn	Member

1.4. Background

The family environment is created human personality from childhood, where the wings for them to life with the joy creative, shimmering colors. The family is the cell of society, the fundamental nucleus of society development, which mounts feelings of all generations of the human race.

Over many generations, the family is formed and Vietnam with the standards developed good values contributes to building the nation's cultural identity. The process of industrialization, modernization and international economic integration creates opportunities and conditions for Vietnam family development, and family and also put family before work difficulties and challenges. Therefore, building family culture and family cultural development is key issue in building work in the cultural life basis, which requires the attention of the Party, the State and society.

Vietnam is a country with a strong culture of Southeast Asia, the generation of the family is an extremely important issue. Vietnamese people think highly of relationship between family members, and want to connect everyone together. Every year, we have relationships and activities tighten together. Therefore, the information of all persons in the same family is extremely important. It helps us to determine and strengthen the relationship, keep in touch and longstanding tradition of the nation.

However, Vietnamese people often have a big family with many generations and the relationship between cousins, brother, sister is very difficult to define. Many forms of greeting go out. Therefore, it’s a big problem for younger people to know and understand about their family, their role in the big family and the way to greeting with other members in the family.

On the other hands, a family tree is a powerful visual way to showcase the structure of your family; make we understanding more clearly and exactly about history of family and make we proud of our family. Whether you have a big family or a small one, you can make a tree that's served as a history lesson and a work of art in one. It could be a documentary for your family's past genealogical data. Depending on the depth you wish to go into, family trees can document hundreds of years back. It’s also excellent for children to learn about their heritage in creative and fun way.

More than that, we can have more our cousin’s information and communicate them easier and faster. Can be told that, family tree is an effective way help us to be closer with our family’s member than ever.

In different case, some members go outside to run business or live in other sites; they often lose contact with others member.

From these, Group FT2013, with Mr.Trần Đình Trí leading, develop the idea of website which collect families’ information from prestigious sources, support advanced search, transverse to user. Then create a family tree.

These Family Tree websites are selected based on 3 factors:

1) **Working well with database**: Because there are too many information about family members, we have to store information in database, select data.
2) **Good-looking design**.
3) **Big community**: We allow user to have more functions and ways to communicate
4) **Fast and exactly website**.

1.5. Literature Review of Existing Systems

1) Genni.com: is a website that allow user to create their family tree. This website provide all most advance functions, but they got 4 issues:

- Provide service for limited user.
- Interface is not attractive
- Cannot traverse from tree to another tree
- Search function cannot show the relationship between node-node exactly.

2) GiaPhaVN 1.0: is a window application. This software is quite good but they still have some problems:

- Cannot display on website interface
- Must download and install
- Interface is not attractive
- Don't have search functions

3) Cây gia phả - author: Nguyễn Văn Tuấn: is a mobile phone application. It specializes for iOS phone. It got some problems too:

- Cannot display on website interface

1.6. Proposal and Expected system

- Allow massive nodes created on Tree with large data and complicated relationships.
- User can search everyone and know the position in the family and find relationship between two people and determine the type of relationship.
- Website use Vietnamese as its main language to help Vietnamese people using more effectively (specialize in people from the North)
- The website helps user to create family tree by drag & drop quickly.
- The website allows user to customize space, color, image and background on their hobbies for the family tree.
- Discriminate people from other gender, position in family, role…
- After create a completely family tree, user can print it to hard copy or PDF, doc.

1.7. Scope

Need to be done:

- Core Function:

+ Construct, Traverse, Add, Edit, Remove node in Tree Structure.

- Database:

+ Minimal use of table, maximize performance for Server when Query and Write Data.

- Website:

+ Design interactive GUI
+ Data Representation
+ Data Transmission Control (to request/submit Data from/to Server)
+ Authority and Security Insurance

- Extend:

+ Print Tree in forms (this is just when everything was done)

No need to be done:

- Apply biology into the system

-Join tree with tree

Chapter 2: Software Project Management Plan

2.1. Problem Definition

2.1.1. Name of this Capstone Project

Family Tree is a system which store information about families, the relationships between cousins and family's members. Family Tree will help users understand more clearly about their family, connect them together, and display the relationship by website interface. Morever, depending on the depth you wish to go into, family trees can document hundreds of years back. It's also excellent for children to learn about their heritage in creative and fun way.

2.1.2. Project Abstract

2.1.2.1. Problem definition

The family environment is created human personality from childhood, where the wings for them to life with the joy creative, shimmering colors. The family is the cell of society, the fundamental nucleus of society development, which mounts feelings of all generations of the human race.

Over many generations, the family is formed and Vietnam with the standards developed good values contributes to building the nation's cultural identity. The process of industrialization, modernization and international economic integration creates opportunities and conditions for Vietnam family development, and family and also put family before work difficulties and challenges. Therefore, building family culture and family cultural development is key issue in building work in the cultural life basis, which requires the attention of the Party, the State and society.

However, Vietnamese people often have a big family with many generations and the relationship between cousins, brother, sister is very difficult to define. Many forms of greetting go out. Therefore, it's a big problem for younger people to know and understand abour their family, their role in the big family and the way to greeting with other members in the family.

In different case, some members go outsite to run business or live in other sites; they often lose contact with others member.

From the belowing problem, our group decides to develop a project to solve these issues named Family Tree.

2.1.2.2. User and their problem

2.1.2.2.1. Family user

It takes a lot of time to know and understand about all of other members in family. Otherwise, greeting really is a big problem if member doesn't know their role well in the family. They can become rule and un-respected.

2.1.2.2.2. Family administrator

The family becomes bigger day by day. Therefore, it's very difficult to manage old or new members.

2.1.2.3. What is the goal?

Develop successfully Family Tree which solve problems in part 2.1 and be used widely.

2.1.2.4. What are the side-effects to be avoided?

Problems can be learned and gain experience from the advantages and disadvantages of existed services:

- Display the Fmaily Tree on Web form
- Advanced search function
- Advanced traverse function
- Good-looking design
- Store data into databse
- Supply advanced authority function

2.1.2.5. Which actions are admissible?

- Firstly, we need find and analyses the advantages and disadvantages of existed systems to learn advantages and avoid disadvantages.
- Then develop new functions to improve our system.

2.1.3. Project Overview

2.1.3.1. The current system:

	Advantages	Disadvantages
Genni.com	- Good-looking design - Have almost needed function: insert, update, delete…	- Provide service for limited user. - Interface is not attractive - Cannot traverse from tree to another tree - Search function cannot show the relationship between node-node exactly.
GiaPhaVN 1.0	- Easy to use	- Cannot display on website interface - Must download and install - Interface is not attractive - Don't have search functions
Cây gia phả - author: Nguyễn Văn Tuấn	- it's a mobile version, stable for mobile device	- Cannot display on website interface - Information that stored in database is not correct.

Base on that table, we try developing complete system which:

- Display the Fmaily Tree on Web form
- Advanced search function
- Advanced traverse function
- Good-looking design
- Store data into databse
- Supply advanced authority function

2.1.3.2. The Proposed System

The final system includes these functions:

- Allow massive nodes created on Tree with large data and complicated relationships.
- User can search everyone and know the position in the family and find relationship between two people and determine the type of relationship.
- Website use Vietnamese as its main language to help Vietnamese people using more effectively (specialize in people from the North)
- The website helps user to create family tree by drag & drop quickly.
- The website allows user to customize space, color, image and background on their hobbies for the family tree.
- Discriminate people from other gender, position in family, role…
- After create a completely family tree, user can print it to hard copy or PDF, doc.

2.1.3.3. Boundaries of the System

System includes:

- The functioning website support many end-devices.
- Final website work well on real environment
- All relevant documents.

2.1.3.4. Development Environment

2.1.3.4.1. Hardware requirement:

- 2 GHz or faster processor
- 2 GB of RAM
- 50 GB (NTFS) of available hard disk space
- 1024 x 768 or higher display resolution
- ADSL internet connection

2.1.3.4.2. Software requirement:

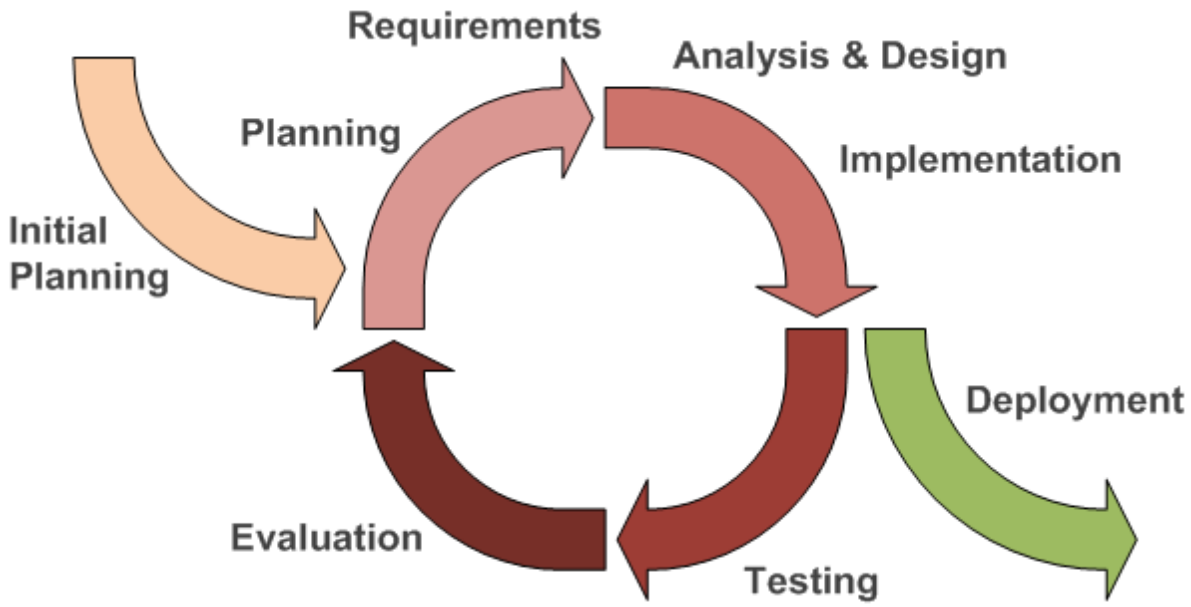
- Windows 7 (x86 or x64) operating system.
- Microsoft Visual Studio for Web 2012
- Microsoft SQL Server 2008R2

2.1.3.4.3. Testing requirement:

- Laptop/PC:
  - Internet connection (DSL+)
  - Browsers: Internet Explorer, Chrome, Firefox

2.2. Project Organization

2.2.1. Software Process Model



The process model chosen for the Family Trees project is “Iterative and Incremental”

2.2.2. Roles and Responsibilities

Name	Role	Responsibilities
Trần Đức Long	- Teamleader - Coder	- Manage project - Study requirement, technique - Design, code
Nguyễn Phương Dung	- Product Quality Assurance/ Tester	- Support PM in manager project - Write requirement - Excuting test - Process Quality Assurance
Phạm Đình Đức	- Developer - Designer	- Config server and environments - Design, code.
Trần Thị Thu Huyền	- Process Quality Assurance/ Tester	- Support PM in manager project - Product Quality Assurance - Excuting test - Study requirement
Nguyễn Thế Trung	- Technical leader - Developer	- Study technique, code

2.2.3. Tools and Techniques

- Development tool: Microsoft Visual Studio for Web 2012
- Database Management System: Microsoft SQL Server 2008R2
- Development Language: ASP.Net, C#
- Others: Microsoft Office 2010, Microsoft Project 2010, Microsoft Visio 2010

2.3. Project Management Plan

2.3.1. Development Phases and Input/output

2.3.1.1 Initiating

Description: kick off meeting, Develop Project Charter and Register Capstone Project

Dependencies and Constraints: N/A

Deliverables: N/A

Resources Needed: LongTD in 2days

Risks: N/A

2.3.1.2 Planning

Description: set plan about time, human resource, extend libraries. Define the project goals and objective.

Dependencies and Constraints: N/A

Output: Introduction and Software Project Management Plan

Deliverables: N/A

Resources Needed: HuyềnTTT, DungNP, and LongTD in 4 days

Risks: conflic ideas and be changed in future.

2.3.1.3 User Specifications Requirement

Description: create User Specifications Requirement

Dependencies and Constraints: N/A

Output: User Specifications Requirement

Deliverables: N/A

Resources Needed: DungNP in 8 days

Risks: conflic or dubplicate ideas

2.3.1.4 Software Requirements Specifications

Description: create Software Requirements Specifications

Dependencies and Constraints: User Specifications Requirement

Output: Software Requirements Specifications

Deliverables: N/A

Resources Needed: DungNP in 10 days

Risks: Structure is wrong or lacking

2.3.1.5 Design Database

Description: create Database

Dependencies and Constraints: SRS document

Output: database diagram and created script

Deliverables: N/A

Resources Needed: LongTD in 7 days

Risks: SRS isn't clear or develop wrong structure

2.3.1.6 GUI design

Description: design template

Dependencies and Constraints: N/A

Output: HTML layout and CSS files

Deliverables: N/A

Resources Needed: ĐứcPĐ in 2 days

Risks: Template don't support all functions

2.3.1.7 Coding

- **Coding GUI**

Description: Code

Dependencies and Constraints: SRS, URD, architecture design, database, layout design.

Output: source code and unit test

Deliverables: N/A

Resources Needed: ĐứcPĐ in 8 days

Risks: issues have come up, having to change requirement and related documents

- **Coding Stored Procedure**

Description: Code

Dependencies and Constraints: SRS, URD, architecture design, database, layout design.

Output: Store procedure

Deliverables: N/A

Resources Needed: LongTD in 11 days

Risks: Lacking store procedure and may be is not correct.

- **Coding tree and logic**

Description: Code

Dependencies and Constraints: SRS, URD, architecture design, database, layout design.

Output: Tree and functions in tree

Deliverables: N/A

Resources Needed: TrungNT (LongTD & DucPD support) in 25 days

Risks: Complex and lost time to find algorithms

2.3.1.8 Testing and Fix bugs

Description: create system testcase and test on local environment

Dependencies and Constraints: SRS and finished source code

Output: Software Test Documentation

Deliverables: N/A

Resources Needed: DungNP, HuyềnTTT, LongTD, TrungNT, ĐứcPĐ in 15 days.

Risks: Bug has come up and takes time to fix, retest.

2.3.1.9 Software User's Manual

Description: write manual

Dependencies and Constraints: SRS and finished source code

Output: Software User's Manual

Deliverables: N/A

Resources Needed: LongTD in 3 days.

2.3.1.10 Deployment

Description: Implement on real environment

Dependencies and Constraints: source code

Output: put system on internet

Deliverables: N/A

Resources Needed: TrungNT in 4 days

Risks: bug come up because conflic between development environment and hosting environment.

2.3.1.11 Submit work

Description: print and burn CD

Dependencies and Constraints: source code, final document

Output: CD and final document

Deliverables: N/A

Resources Needed: TrungNT, DucPD in 1 day

Risks: issue in printing or burning CD

2.3.2. TaskSheet

Family Tree_Time_Resource

Task Name	Duration	Start	Finish	Predecessors	Resource Names
Family Tree System	**81 days**	**Mon 5/13/13**	**Sat 8/31/13**		
1. Initiating	**2 days**	**Mon 5/13/13**	**Tue 5/14/13**		
1.1. Kickoff meeting	3 hrs	Mon 5/13/13	Mon 5/13/13		LongTD
1.2. Develop project charter	1 hr	Mon 5/13/13	Mon 5/13/13		LongTD
1.3. Register Capstone Project	1 day	Tue 5/14/13	Tue 5/14/13		LongTD
2. Planning	**4 days**	**Wed 5/15/13**	**Mon 5/20/13**		
2.1. Writing Introduction	3 days	Wed 5/15/13	Fri 5/17/13		HuyenTTT
2.2. Review Introduction	3 hrs	Sat 5/18/13	Sat 5/18/13		DungNP
2.3. Writing Software Project Management Plan	1.5 days	Sat 5/18/13	Mon 5/20/13		DungNP
2.4. Review Software Project Management Plan	0.5 days	Mon 5/20/13	Mon 5/20/13		LongTD
3. User Specifications Requirement	**8 days**	**Tue 5/21/13**	**Thu 5/30/13**		
3.1. Writing User Specifications Requirement	7 days	Tue 5/21/13	Wed 5/29/13		DungNP
3.2. Review User Specifications Requirement	1 day	Thu 5/30/13	Thu 5/30/13		DungNP
4. Mid-term review	**5 days**	**Fri 5/31/13**	**Thu 6/6/13**		
4.1. Mid-term review documents	3 days	Fri 5/31/13	Tue 6/4/13		HuyenTTT
4.2. Finish changes	2 days	Tue 6/4/13	Wed 6/5/13		HuyenTTT
5. Software Requirements Specifications	**10 days**	**Thu 6/6/13**	**Wed 6/19/13**		
5.1. Writing Software Requirements Specifications	8 days	Thu 6/6/13	Mon 6/17/13		DungNP
5.2. Review Software Requirements	2 days	Tue 6/18/13	Wed 6/19/13		DungNP

Specifications					
6. Developing	**30 days**	**Wed 6/19/13**	**Tue 7/30/13**		
6.1. Design GUI	2 days	Wed 6/19/13	Thu 6/20/13		DucPD
6.2. Design database	7 days	Fri 6/21/13	Mon 7/1/13		LongTD
6.3. Code Demo for main function	3 days	Fri 6/21/13	Tue 6/25/13		TrungNT
6.4. Coding GUI	8 days	Wed 6/26/13	Fri 7/5/13		DucPD
6.5. Coding Stored procedure	11 days	Wed 6/26/13	Wed 7/10/13		LongTD
6.6. Coding tree and Logic	25 days	Wed 6/26/13	Tue 7/30/13		TrungNT
7. Testing and fix bug	**15 days**	**Mon 7/22/13**	**Fri 8/9/13**		
7.1. Write system test case	5 days	Mon 7/22/13	Fri 7/26/13		HuyenTTT
7.2. Execute system test	8 days	Sat 7/27/13	Tue 8/6/13		DungNP
7.3. Fix bugs	8 days	Sat 7/27/13	Tue 8/6/13		DucPD,LongTD,TrungNT
7.4. Complete Report 1, 2, 3, 4	2 days	Wed 8/7/13	Thu 8/8/13		DungNP
8. Closing	**16 days**	**Sat 8/10/13**	**Fri 8/30/13**		
8.1. Writing Software Test Documentation	3 days	Sat 8/10/13	Tue 8/13/13		HuyenTTT
8.2. Writing Software User's Manual	3 days	Wed 8/14/13	Fri 8/16/13		LongTD
8.3. Revise final document	3 days	Sat 8/17/13	Tue 8/20/13		HuyenTTT
8.4. Deployment	4 days	Wed 8/21/13	Mon 8/26/13		TrungNT
8.5. Burning CD & Print	1 day	Tue 8/27/13	Tue 8/27/13		DungNP
8.6. Project presentation	3 days	Wed 8/28/13	Fri 8/30/13		LongTD,DungNP,TrungNT,DucPD,HuyenTTT

2.3.3. All Meeting Minutes

MeetingMinutes_Family Tree..xls

Chapter 3: Software Requirements Specifications

3.1. User Requirement Specification

3.1.1 Overview

Family Tree is a web application that enables peoples to easily and efficiently view structure of family, create a tree or search relationship between two people in a short period of time. Family Tree is available to all the people at no cost and provides a secure environments for family administrator, family user to interact online according to their role in a nearly real-time environment.

3.1.2 Functional Requirements

To sum it all up, our system has 3 roles as System administrator, Family administrator and Family user and functions are matched respectively as below:

- System Administrator: Administrator the Family Tree users which include Family Administrator and Family user and their access to the system as well as manage user account and tree.
- Family Administrator: who can administer the Family users and their access to tree for their corresponding family
- Family user: who has permission to view tree for their corresponding family but not Create/Edit/Delete.

Based on roles, we have partitioned into sub functionalities as table below:

ID	Functionality	Sub functionality	Use cases	Descriptions
1	System Administrator functions	Login	UC-001	
2		View user list	UC-002	
3		Edit user account	UC-003	
4		Approve family administrator	UC-004	
5		Delete user account	UC-005	
6		Delete tree	UC-006	
7		Restore a tree	UC-007	
8		Log Out	UC-008	
9	Family Administrator functions	Create account	UC-009	
10		Login	UC-010	
11		Approve family users	UC-011	
12		Create tree	UC-012	
13		View/Edit node	UC-013	
14		Delete node	UC-014	
15		Expanding Tree	UC-015	

16		Search relationship between 2 peoples in a tree	UC-016	
17		Print a tree	UC-017	
18		Customize tree	UC-018	
19		Notification: review comment from family user	UC-019	
20		Log Out	UC-020	
21	Family user functions	Create Account	UC-021	
22		Login	UC-022	
23		View a tree	UC-023	
24		Search relationship between 2 peoples in a tree	UC-024	
25		Print tree	UC-025	
26		Expanding Tree	UC-026	
27		Notification: send comment to Family Administrator	UC-027	
28		Log Out	UC-028	

3.2. System Requirement Specification

3.2.1 External Interface Requirements

3.2.1.1 User Interfaces

3.2.1.2 Hardware Interfaces

- Family Tree system has been developed in C# programming language for ASP.Net Web form on the .NET framework. It requires supported .Net Framework 4.0 Windows hosting and Microsoft SQL Server 2008 or better.
- Website support all famous browsers on PC, laptop like Internet Explorer, Chrome with internet connection.

3.2.1.3 Software Interfaces

- Family Tree developed # programming language for ASP.Net Web form on the .NET framework 4.0.
- Database is stored on Microsoft SQL 2008.
- Tool for development are Visual Studio for Web 2012.

3.2.1.4 Communications Protocol

- Family Tree system will use HTTP protocol for communication over internet and connect Internet to load family tree information.

3.2.2 Use case

3.2.2.1 Family Tree use-case diagram

3.2.2.2 System Administrator functions

3.2.2.2.1 [System Administrator]Login

USE CASE – Log in as System Administrator Specification			
Use-case No.	UC001	**Use-case Version**	1.0
Use-case Name	Log in to the Family Tree system by System Administrator		

Author	Dung Nguyen		
Date	18th May 2013	Priority	High

Actor:

- System administrator

Summary:

- Log in to the Family Tree system by System Administrator role

Goal:

- This use case allows System Administrator to log in to the Family Tree system and view the starting (Home) page and the system administrator functions.

Triggers:

- This use case is invoked when the user navigates to the Family Tree URL.

Preconditions:

- The user attempts to log in to the Family Tree system.

Post Conditions:

- The user has logged in to the Family Tree system or has been denied access.
- User is able to successfully change his password
- User is able to request help with a forgotten user ID.
- User is able to request that password be sent when password is forgotten.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Navigate to the Family Tree URL	Display the Family Tree Sign On page containing the following information: • Family Tree informational text (if any)(future batch) • User Name (required) • Password (required) • Create My Family Tree! button • Join in My Family Tree! button • Forgot your user name? (as a link) • Forgot your password? (as	

		a link) • Login button	
2	Enter the User Name and Password and choose an option to Log in	If no errors occur, display the Home Page containing the following information: • User Name of current user • Menu options, based on the permissions of the user • Links for Family Tree Home page, My Account option and Logout option	

Alternative Scenario:

ID	User action	System Response	Comment
1	Navigate to the Family Tree URL	Display the Family Tree Sign On page containing the following information: • Family Tree informational text (if any)(future batch) • User Name (required) • Password (required) • Create My Family Tree! button • Join in My Family Tree! button • Forgot your user name? (as a link) • Forgot your password? (as a link) • Login button	
2	Enter the User Name and Password and choose an option to Log in	If the User Name and Password are not valid, display the message "User Name and Password are not valid."	

Exceptions:

ID	User Action	System Response	Comment
1	Navigate to the Family Tree URL	Display the Family Tree Sign On page containing the following information: • Family Tree informational text (if any)(future batch) • User Name (required) • Password (required) • Create My Family Tree! button • Join in My Family Tree! button • Forgot your user name? (as a link) • Forgot your password? (as a link) • Login button	
2	Enter the User Name and Password and choose an option to Log in	Validate the User Name and Password. If the User Name or Password field is blank, display the message "Both User Name and Password are required."	

Relationships: N/A

Business Rules:

- If user name or password is blank, back to Log in page and show error message.
- If user name or password is not match, back to Log in page and show error message.

Data definition- Log in page

ID	Field name	Control Type	Data type	Required?	Note
Log in section					
1	User name	Text box	String(255)	Y	• Email address or User ID of the user • Must be a valid existing user

2	Password	Text box	String (255)	Y	• Password of the user • Displayed as a masked value • Must be valid for the User Name
Control and Links					
1	Log in	Command button	N/A	N/A	
2	Create My Family tree!	Command button	N/A	N/A	
3	Join in My Family Tree!	Command button	N/A	N/A	
4	Forgot your user name?	Link	N/A	N/A	
5	Forgot your password?	Link	N/A	N/A	

3.2.2.2.2 [System Administrator] View/edit user

USE CASE – View/Edit Specification			
Use-case No.	UC002 + UC003	**Use-case Version**	1.0
Use-case Name	View/Edit user		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

System administrator

Summary:

- View and edit user accounts for access to the Family Tree system.

Goal:

- View and edit user accounts for access to the Family Tree system

Triggers:

- This use case is invoked when the user logs in to the Family Tree system

as a System Administrator.

Preconditions:

- The user must be logged into the Family Tree system with System Administrator permission.

Post Conditions:

- One or more Family Tree user accounts have been viewed, edited and/or removed from a family

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has User Administrator role	If no errors occur, display the System Administrator Home Page containing the following information: • User Name of current user • Menu options, based on the permissions of the user • Links for Family Tree Home page, My Account option and Logout option	
2	Select an option to Manage User Accounts	Displays a list of Family Tree users containing the following User Information for each user in a list of users, sorted by Login ID(ascending): • Login ID (Username+ Password) • Email • Address • Actions: Edit, Delete, Approve	
3	Optionally, choose the [Edit] action and makes additional changes and chooses to Save	+ If Save is chosen and any required field does not contain a value, displays the message "Please enter all required fields." and does not save the changes.	

		+ If Save is chosen and an error does not occur, saves the changes and redisplays the list of users as in step 2 of this flow.	
4	Optionally makes additional changes and chooses to Cancel the changes.	Cancel is chosen, the message shows: "There are unsaved changes. Are you sure you want to cancel anyway?" with OK / Cancel options - Select OK to discard changes and go to destination page - Select Cancel to stay on the current page with all changes and not save	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.2.3 [System Administrator] Approve Family Administrator

USE CASE – Approve Family Administrator Specification			
Use-case No.	UC004	**Use-case Version**	1.0
Use-case Name	Approve Family Administrator		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

System administrator

Summary:

- Approve family administrator for access to the Family Tree system.

Goal:

- Approve family administrator for access to the Family Tree system

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a System Administrator.

Preconditions:

- The user must be logged into the Family Tree system with System Administrator permission.

Post Conditions:

- One or more Family Administrator has been approved.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has User Administrator role	If no errors occur, display the System Administrator Home Page containing the following information: • User Name of current user • Menu options, based on the permissions of the user • Links for Family Tree Home page, My Account option and Logout option	
2	Select an option to Approve family administrator	Approve the family administrator	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.2.4 [System administrator]Delete user account

USE CASE – Delete User Account Specification

Use-case No.	UC004	**Use-case Version**	1.0
Use-case Name	Delete user account		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

System administrator

Summary:

- Delete user account which has been created by Family Administrators

Goal:

- Delete a user account which has been created by Family Administrators

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a System Administrators

Preconditions:

- The user must be logged into the Family Tree system with System Administrators permission.

Post Conditions:

- One or more user account has been removed.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has User Administrators role Go to Manage User account, highlight trees that user can do the deletion Select [Delete] button	Display the confirmation popup that contains: • "Are you sure you want to delete this user?" message with 2 options: Yes/No	
2	Choose [Yes] button	• Delete the user by updating IsDeletedFlag to 1 • Refresh Manage User Account page	
3	Alternatively, choose [No]	Terminate the deletion. No	

	button	change to the user data	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.2.5 [System Administrator] Delete tree

USE CASE – Delete Tree Specification			
Use-case No.	UC006	**Use-case Version**	1.0
Use-case Name	Delete tree		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

System administrator

Summary:

- Delete a tree which has been created by Family Administrators

Goal:

- Delete a tree which has been created by Family Administrators

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a System Administrators

Preconditions:

- The user must be logged into the Family Tree system with System Administrators permission.

Post Conditions:

- One or more Family Tree has been removed.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has User	Display the confirmation popup that contains:	

	Administrators role Go to Manage Tree, highlight trees that user can do the deletion Select [Delete] button	• "Are you sure you want to delete this item?" message with 2 options: Yes/No	
2	Choose [Yes] button	• Delete the tree by updating IsDeletedFlag to 1 • Refresh Manage Tree page, with is deleted tree? = [Yes]	
3	Alternatively, choose [No] button	Terminate the deletion. No change to the tree data	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.2.6 [System Administrator] Restore tree

USE CASE –Restore a tree Specification			
Use-case No.	UC007	**Use-case Version**	1.0
Use-case Name	Restore tree		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	Medium

Actor:

System administrator

Summary:

- Restore a tree which has been deleted before by System Administrator or Family Administrators

Goal:

- Restore a tree which has been deleted by System Administrators or

Family Administrators

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a System Administrators

Preconditions:

- The user must be logged into the Family Tree system with System Administrators permission.

Post Conditions:

- One or more Family Tree has been restored.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has User Administrator role	Login system Family Tree Administrator successfully It contains 2 options in the menu bars: Manage User Account, Manage Tree	
2	Choose Manage Tree option from the main menu	• System displays the tree list and has the following items: • - Deleted' column (either 'Yes' or [blank]) is added at the far right of the column order (see wire frame) (future batch). • - Either 'Restore' or 'Administrator Delete' actions will appear based on the value in the 'Deleted' field: • + If 'Deleted' = Yes => Show 'Restore' as available + If 'Deleted' = '[Blank]' => Show 'Administrator Delete' as available	
3	User searches for the tree (has already been deleted; 'Deleted' = Yes)	System displays the tree that has been deleted.	
4	With the tree highlighted, the	System displays the message:	

	user selects the 'Restore' action	"Are you sure you want to restore the previously deleted tree?" with 2 options: Yes/No	
5	Choose [Yes] option	System performs a 'restore' of the tree, which means: - 'Deleted' column is set to '[Blank]' on the tree. - Family User and Family administrator with access to the work item will see it reappear in tree list.	
6	Alternatively, choose [No] button	The tree is not restored.	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.2.7 [System Administrator] Log Out

USE CASE – Delete User Account Specification			
Use-case No.	UC008	**Use-case Version**	1.0
Use-case Name	Log Out		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	Low

Actor:

System administrator

Summary:

- Log out Family Tree system

Goal:

- System administrator log out successfully.

Triggers:

- This use case is invoked when the user logs in to the Family Tree system

as a System Administrators

Preconditions:

- The user must be logged into the Family Tree system with System Administrators permission.

Post Conditions:

- Log out Family Tree system

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has User Administrators role and is in any page of Family Tree system Choose [Log out] hyperlink	Log out Family Tree system	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.3 Family Administrator functions

3.2.2.3.1 [Family Administrator] Create account & Log into Family Tree system

USE CASE –Family Administrator- Create account Specification			
Use-case No.	UC009 + UC010	**Use-case Version**	1.0
Use-case Name	Family Administrator – Create account & Log into Family Tree system		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

- Family administrator

Summary:

- Create Family Administrator account in Family Tree system
- Log into Family Tree system as Family Administrator role

Goal:

- This use case allows a user to create a Family Administrator account to log in to the Family Tree system and view the starting (Home) page and the family administrator functions.

Triggers:

- This use case is invoked when the user navigates to the Family Tree URL.

Preconditions:

- The user attempts to Login page of the Family Tree system.

Post Conditions:

- User is able to successfully create his account and login.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Navigate to the Family Tree URL	Display the Family Tree Sign On page containing the following information: • Family Tree informational text (if any)(future batch) • User Name (required) • Password (required) • Create My Family Tree! button • Join in My Family Tree! button • Forgot your user name? (as a link) • Forgot your password? (as a link) • Login button	
2	Click on Create My Family Tree! button	Display the Create a new Family Tree account page containing the following information: • User name (required) • Password (required) • Confirm password (required) • Email address • My Family Tree information section: + My Family Tree name:	

		(required) + Address (required) • Create an account button(in blue color)	
	Choose Create an account button	If no error occurs, displays message: Create account successfully.	

Alternative Scenario:

ID	User action	System Response	Comment
1	Navigate to the Family Tree URL	Display the Family Tree Sign On page containing the following information: • Family Tree informational text (if any)(future batch) • User Name (required) • Password (required) • Create My Family Tree! button • Join in My Family Tree! button • Forgot your user name? (as a link) • Forgot your password? (as a link) • Login button	
2	Enter the User Name and Password and choose an option to Log in	Display the error message: You have not account. Please click on Create My Family Tree! Or Join in My Family Tree! Button to create your account.	

Exceptions:

Relationships: N/A

Business Rules:

Data definition- Log in page

ID	Field name	Control Type	Data type	Required?	Note
Log in section					
1	User name	Text box	String(255)	Y	• Email address or User ID of the user • Must be a valid existing user
2	Password	Text box	String (255)	Y	• Password of the user • Displayed as a masked value • Must be valid for the User Name
Control ands Links					
1	Log in	Command button	N/A	N/A	
2	Forgot your user name?	Link	N/A	N/A	
3	Forgot your password?	Link	N/A	N/A	
4	Create My Family Tree	Command button	N/A	N/A	
5	Join in My Family Tree	Command button	N/A	N/A	

Data definition – Create My Account page

ID	Field name	Control Type	Data type	Required?	Note
User information					
1	User name	Text box	String(255)	Y	• Email address or User ID of the user • Must be a valid existing user
2	Password	Text box	String (255)	Y	• Password of the user • Displayed as a masked value • Must be valid for the User Name

3	Confirm password	Text box	String(255)	Y	• Password of the user • Displayed as a masked value Must be valid for the User Name
My Family Information					
1	My Family name	Text box	String (255)	Y	
2	Address	Text box	String(255)	Y	
3					
Control and Links					
1	Create an account	Command button	N/A	N/A	

3.2.2.3.2 [Family Administrator] Approve Family users

USE CASE – Approve Family Users Specification			
Use-case No.	UC004	**Use-case Version**	1.0
Use-case Name	Approve Family Users		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

Family administrator

Summary:

- Approve family users for access to his family.

Goal:

- Approve family users for access to his family.

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a System Administrator.

Preconditions:

- The user must be logged into the Family Tree system with System Administrator permission.

Post Conditions:

- One or more Family users has been approved.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family Administrator role	If no errors occur, display the Family Administrator Home Page containing the following information: • User Name of current user • Menu options, based on the permissions of the user • Links for Family Tree Home page, My Account option and Logout option	
2	Select an option to Approve family users	Approve the family users.	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.3.3 [Family Administrator] Create tree

USE CASE –Add new node Specification			
Use-case No.	UC012	**Use-case Version**	1.0
Use-case Name	Create Tree		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High
Actor: Family administrator			

Summary:

- Create new tree via Family Tree system

Goal:

- Create new tree via Family Tree system

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family Administrator

Preconditions:

- The user must be logged into the Family Tree system with Family Administrator

Post Conditions:

- A family tree has been created successfully.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family Administrator role Select an option to Create new Tree	Displays a add new tree page containing the following as default: • Current user's node (Last name+ Fist name) • Add my father's node at the upper left of current user's node • Add my mother's node at the upper right of current's user node. • Right arrow next to current user's node to add brother, sister, husband (If current user is female), wife(If current user is male), Cancel button • Down arrow next to current user's node to add son or add daughter, Cancel button	
2	User click on Add My Father or Add My Mother node Alternatively, the user can click on 2 arrow (right or	Display the Add <Mother>(or other) pop up containing the following information:	

	down arrows) to Add Bother, Add Sister, Add Husband (or Add Wife),Add Son or Add Daughter	• Title: < Mother>, <Father>,… • First name (required) • Last name (required) • Living or Deceased () • More field hyperlink • Cancel button (in grey color) • Add button (in blue color)	
	User click on More field hyperlink at the bottom of Add <Mother> pop up	Display fully the Adding mother pop up containing the following information: • Title: Adding <Mother>… • First name • Middle name (optional) • Lat name • Birth Surname (optional) • Living or Deceased • Email address(optional) • Date of birth • Place of birth • Current location • Other	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.3.4 [Family Administrator] View/edit a node

USE CASE –View/Edit node Specification			
Use-case No.	UC013	**Use-case Version**	1.0

Use-case Name	View/Edit a node in a Family Tree		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

Family administrator

Summary:

- View/Edit a node in a family tree

Goal:

- View/Edit a node in a family tree

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family Administrator

Preconditions:

- The user must be logged into the Family Tree system with Family Administrator

Post Conditions:

- One or more node in Family Tree have been viewed, edited from a family tree

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family Administrator role Select an option to Create new Tree	Displays a add new tree page containing the following as default: • Current user's node (Last name+ Fist name) • Add my father's node at the upper left of current user's node • Add my mother's node at the upper right of current's user node. • Right arrow next to current user's node to add brother, sister, husband (If current user is female), wife(If current user is male), Cancel button • Down arrow next to current user's node to add son or add daughter,	

		Cancel button • Edit hyperlink • Add photo of person (placed in a square)	
2	User click on Edit hyperlink at the bottom of person information area	Display the Add <Mother>(or other) pop up containing the following information: • Title: < Mother>, <Father>,… • Name: First name, Middle name, Last name, Suffix • Display name • Birth surname • As known As: separate nick names with a comma • Status: Living or Deceased • Gender: Male, Female • Add email • Occupation • Current Location: • Edit current Location hyperlink: + Place name: + City: + County: + State/Province: + Country: • Date of birth with 4 options: + Exact: MM/DD/YYYY + After: MM/DD/YYYY + Before: MM/DD/YYYY +Between: From…To..(MM/DD/YYYY) • Save and changes button (in blue color)	

		• Cancel button (in grey color)	
	Make the change, and then click on Save & Change button	Save the changes and back to tree	
	Make the change, and then click on Cancel button	Cancel the change and back to tree	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.3.5 [Family Administrator] Delete node

USE CASE –Delete a node Specification			
Use-case No.	UC014	**Use-case Version**	1.0
Use-case Name	Delete node		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

Family administrator

Summary:

- Delete a node in a Family Tree

Goal:

- Delete a node in a Family Tree

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family Administrator

Preconditions:

- The user must be logged into the Family Tree system with Family Administrator

Post Conditions:

- One or more node in Family Tree have been viewed, deleted from a family tree

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family Administrator role	Displays a family tree that has been created. The Delete button (x) is located in the upper right side of each node.	
2	Chooses an option to Delete the node	Displays the "Remove person from your tree" confirmation message: "Are you sure you would like to remove this person from your tree?" and allows the user to choose Yes or No.	
3	Chooses Yes.	+ If the node can be deleted: • Deletes information of this node from the database • Removes the node from your tree Saves the changes.	
4	Choose No.	Closes the confirmation message and does not delete the node.	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.3.6 [Family Administrator] View/Expanding Tree

USE CASE – Expanding Tree Specification			
Use-case No.	UC015	**Use-case Version**	1.0
Use-case Name	View/Expanding Tree		

Author	Dung Nguyen		
Date	18th May 2013	**Priority**	Medium

Actor:

Family administrator

Summary:

- Expanding a tree at both the front and the rear of the tree

Goal:

- Expanding a tree at both the front and the rear of the tree

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family Administrator

Preconditions:

- The user must be logged into the Family Tree system with Family Administrator

Post Conditions:

- A family tree can be expand at both the front and the rear of the tree

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family Administrator role	Displays a family tree that has been created. + Up arrow used to see the top of part of the family tree. + Down arrow used to see the bottom of part of the family tree	
2	Click on Up arrow Note that: Up arrow just appear when user point mouse pointer at the top of part of the family tree	Display the top of part of the family tree	
3	Click on Down arrow Note that: Down arrow just appear when user point mouse pointer at the bottom	Display the bottom of part of the family tree	

	of part of the family tree		

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.3.7 [Family Administrator] Search relationship between 2 persons in a tree or 2 trees

USE CASE –Search person Specification			
Use-case No.	UC016	**Use-case Version**	1.0
Use-case Name	Search relationship between 2 persons in a tree or 2 trees		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

Family administrator

Summary:

- Search relationship between 2 persons in a tree or 2 trees

Goal:

- Search relationship between 2 persons in a tree or 2 trees

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family Administrator

Preconditions:

- The user must be logged into the Family Tree system with Family Administrator

Post Conditions:

- Relationships between 2 persons in a Family Tree have been found from a family tree or two family trees.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family Administrator role	Displays a family tree that has been created.	

2	Choose the Search button	The system displays the Search pop up which contains the following informations: + Person 1's name + Person 1's family + Person 1's father + Person 1's grandfather + Person 2's name + Person 2's family + Person 2's father + Person 2's grand father + Search/Cancel buttons	
3	User inputs search information for 2 persons, then click the [Search] button	Display result in the screen (relationship between 2 persons)	
4	Alternatively, user may choose [Cancel] button	+ Closes Search pop up	

Alternative Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family Administrator role	Displays a family tree that has been created.	
2	User selects the first person and the second person (in the same family tree or different)	The system highlights the way from the first person to the second person and display result in the screen.	

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.3.8 [Family Admin] Print tree

USE CASE –Review comment from family user Specification			
Use-case No.	UC018	**Use-case Version**	1.0
Use-case Name	Print		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	Low

Actor:

Family administrator

Summary:

- Family administrator prints a family tree to PDF format

Goal:

- Family administrator prints a family tree to PDF format

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family Administrator

Preconditions:

- The user must be logged into the Family Tree system with Family Administrator

Post Conditions:

- Family administrator can print a family tree to PDF format

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family Administrator role	• Displays a family tree that has been created.	
2	User chooses [Print] button	• Save the family tree in PDF format	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.3.9 [Family Administrator] Review comment from family user

USE CASE –Review comment from family user Specification			
Use-case No.	UC019	**Use-case Version**	1.0
Use-case Name	Review comment from family user		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	Medium

Actor:

Family administrator

Summary:

- Review comment from family user

Goal:

- Review comment from family user

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family Administrator

Preconditions:

- The user must be logged into the Family Tree system with Family Administrator

Post Conditions:

- Family administrator can review comment from family user.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family Administrator role	• The system will automatically notify the Family Administrator, upon login, if any new pending comments are available. • Displays a family tree that has been created.	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

BR	Description	Note
1	Family Administrator cannot create a New Comment, only review comment from family user	
2	Family Administrator will be able to Reply to a message received from a family user (one time only).	
3	+ Family administrator will be able to View All Comments in reverser date order (newest to oldest) as a way to review Comments that have already been marked as "Read". + Once a comment is marked as "Read" by the Family Administrator, it is no longer in pending notification status. -	

3.2.2.3.10 [Family Administrator] Log out

USE CASE – Delete User Account Specification			
Use-case No.	UC020	**Use-case Version**	1.0
Use-case Name	Log Out		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	Low

Actor:

Family administrator

Summary:

- Log out Family Tree system

Goal:

- Family administrator log out Family Tree successfully.

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family Administrators

Preconditions:

- The user must be logged into the Family Tree system with Family Administrators permission.

Post Conditions:

- Log out Family Tree system

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who hasFamily Administrators role and is in any page of Family Tree system Choose [Log out] hyperlink	Log out Family Tree system	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.4 Family user functions

3.2.2.4.1 [Family user]Create account and Log into Family Tree system

USE CASE –Family Administrator- Create account Specification			
Use-case No.	UC021 + UC022	**Use-case Version**	1.0
Use-case Name	Family User – Create account and Login		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

- Family user

Summary:

- Create Family user account in Family Tree system
- Log in Family Tree system.

Goal:

- This use case allows a user to create a Family user account to log in to the Family Tree system and view the starting (Home) page and the family user functions.

Triggers:

- This use case is invoked when the user navigates to the Family Tree URL.

Preconditions:

- The user attempts to Login page of the Family Tree system.

Post Conditions:

- User is able to successfully create his account.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Navigate to the Family Tree URL	Display the Family Tree Sign On page containing the following information: • Family Tree informational text (if any)(future batch) • User Name (required) • Password (required) • Create My Family Tree! button • Join in My Family Tree! button • Forgot your user name? (as a link) • Forgot your password? (as a link) • Login button	
2	Click on Join in My Family Tree! button	Display the Create a new Family Tree account page containing the following information: • User name (required) • Password (required) • Confirm password (required) • Email address • Select your family: display all family tree that has been created by Family Administrator • Create an account button(in blue color)	
3	Choose Create an account	If no error occurs, displays	

	button	message: Create account successfully.	

Alternative Scenario:

ID	User action	System Response	Comment
1	Navigate to the Family Tree URL	Display the Family Tree Sign On page containing the following information: • Family Tree informational text (if any)(future batch) • User Name (required) • Password (required) • Create My Family Tree! button • Join in My Family Tree! button • Forgot your user name? (as a link) • Forgot your password? (as a link) • Login button	
2	Enter the User Name and Password and choose an option to Log in	Display the error message: You have not account. Please click on Create My Family Tree! Or Join in My Family Tree! Button to create your account.	

Exceptions:

Relationships: N/A

Business Rules:

Data definition- Log in page

ID	Field name	Control Type	Data type	Required?	Note
Log in section					
1	User name	Text box	String(255)	Y	• Email address or User ID of the user • Must be a valid existing

					user
2	Password	Text box	String (255)	Y	• Password of the user • Displayed as a masked value • Must be valid for the User Name
Control ands Links					
1	Log in	Command button	N/A	N/A	
2	Forgot your user name?	Link	N/A	N/A	
3	Forgot your password?	Link	N/A	N/A	
4	Create My Family Tree	Command button	N/A	N/A	
5	Join in My Family Tree	Command button	N/A	N/A	

Data definition – Create My Account page

ID	Field name	Control Type	Data type	Required?	Note
User information					
1	User name	Text box	String(255)	Y	• Email address or User ID of the user • Must be a valid existing user
2	Password	Text box	String (255)	Y	• Password of the user • Displayed as a masked value • Must be valid for the User Name
3	Confirm password	Text box	String(255)	Y	• Password of the user • Displayed as a masked value Must be valid for the User Name
4	Email	Text box	String(255)	N	• Email must be in email

					format: abcxyz@abc.xyz
5	Select your family tree	Combo box		Y	• Display all family tree that has been created by family Administrator
Control and Links					
1	Create an account	Command button	N/A	N/A	

3.2.2.4.2 [Family user] View a family tree

USE CASE – View a tree Specification			
Use-case No.	UC023	**Use-case Version**	1.0
Use-case Name	View a family tree		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

Family user

Summary:

- View tree which has been created by Family users

Goal:

- View a tree which has been created by Family users

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a System users

Preconditions:

- The user must be logged into the Family Tree system with System users

Post Conditions:

- One or more Family Tree has been viewed.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family user role	Open that tree, and System administrator can view the tree, but cannot edit anything	

	Go to Manage Tree, highlight trees, select [View] button	in tree	
2	Alternatively, the system administrator can choose [Log Out] option in menu bar	Log out from tree	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.4.3 [Family Admin] Print tree

USE CASE –Review comment from family user Specification			
Use-case No.	UC024	**Use-case Version**	1.0
Use-case Name	Print		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	Low

Actor:

Family user

Summary:

- Family user prints a family tree to PDF format

Goal:

- Family user prints a family tree to PDF format

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family user

Preconditions:

- The user must be logged into the Family Tree system with Family user

Post Conditions:

- Family administrator can print a family tree to PDF format

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system	• Displays a family tree that	

	by user who has Family user role	has been created.	
2	User chooses [Print] button	• Save the family tree in PDF format	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.4.4 [Family Administrator] Search relationship between 2 persons in a tree or 2 trees

USE CASE –Search person Specification			
Use-case No.	UC025	**Use-case Version**	1.0
Use-case Name	Search relationship between 2 persons in a tree or 2 trees		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

Family user

Summary:

- Search relationship between 2 persons in a tree or 2 trees

Goal:

- Search relationship between 2 persons in a tree or 2 trees

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family user

Preconditions:

- The user must be logged into the Family Tree system with Family user

Post Conditions:

- Relationships between 2 persons in a Family Tree have been found from a family tree or two family trees.

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system	Displays a family tree that has	

	by user who has Family user role	been created.	
2	Choose the Search button	The system displays the Search pop up which contains the following informations: + Person 1's name + Person 1's family + Person 1's father + Person 1's grandfather + Person 2's name + Person 2's family + Person 2's father + Person 2's grand father + Search/Cancel buttons	
3	User inputs search information for 2 persons, then click the [Search] button	Display result in the screen (relationship between 2 persons)	
4	Alternatively, user may choose [Cancel] button	+ Closes Search pop up	

Alternative Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family user role	Displays a family tree that has been created.	
2	User selects the first person and the second person (in the same family tree or different)	The system highlights the way from the first person to the second person and display result in the screen.	

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.4.5 [Family user] Expanding tree

USE CASE – Expanding Tree Specification			
Use-case No.	UC026	**Use-case Version**	1.0
Use-case Name	Expanding Tree		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

Family user

Summary:

- Expanding a tree at both the front and the rear of the tree

Goal:

- Expanding a tree at both the front and the rear of the tree

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family user

Preconditions:

- The user must be logged into the Family Tree system with Family user

Post Conditions:

- A family tree can be expand at both the front and the rear of the tree

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family user role	Displays a family tree that has been accessed. + Up arrow used to see the top of part of the family tree. + Down arrow used to see the bottom of part of the family tree	
2	Click on Up arrow Note that: Up arrow just appear when user point mouse pointer at the top of	Display the top of part of the family tree	

	part of the family tree		
3	Click on Down arrow Note that: Down arrow just appear when user point mouse pointer at the bottom of part of the family tree	Display the bottom of part of the family tree	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.2.2.4.6 [Family user] Send comment to family admin

USE CASE –Send comment to family administrator Specification			
Use-case No.	UC027	**Use-case Version**	1.0
Use-case Name	Send comment to family administrator		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	High

Actor:

Family user

Summary:

- Send comment to family administrator

Goal:

- Send comment to family administrator

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family user

Preconditions:

- The user must be logged into the Family Tree system with Family User

Post Conditions:

- Send comment to family administrator

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system	• Displays a family tree that	

	by user who has Family Administrator role	has been created. • Send comment button is located at the upper right side of family tree	
2	Invokes the "Comment" action	• The system prompts the user for text comments.	
3	Choose the family admin, enter comment and invokes the [Send] action	• Comment has been sent.	
	Alternatively, user can choose the [Cancel] button	Do not send comment to family administrator.	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

BR	Description	Note
1	The Family user can select an individual to send comments to from a drop-down list + The drop-down list will be limited to the family administrator from the facility the family user is associated with. + If only one family administrator is available for the facility that the family user is associated with, the system should just default to that user.	
2	+ The Family User can view a Reply from an Family Administrator. + The Family User cannot Reply to a Replied Comment from an Family administrator.	
3	The Family User can not send comment to an family administrator in other family.	

3.2.2.4.7 [Family user] Log Out

USE CASE – Delete User Account Specification			
Use-case No.	UC028	**Use-case Version**	1.0
Use-case Name	Log Out		
Author	Dung Nguyen		
Date	18th May 2013	**Priority**	Low

Actor:

Family user

Summary:

- Log out Family Tree system

Goal:

- Family user log out successfully.

Triggers:

- This use case is invoked when the user logs in to the Family Tree system as a Family user

Preconditions:

- The user must be logged into the Family Tree system as Family user

Post Conditions:

- Log out Family Tree system

Main Success Scenario:

ID	User Action	System Response	Comment
1	Login Family Tree system by user who has Family user and is in any page of Family Tree system Choose [Log out] hyperlink	Log out Family Tree system	

Alternative Scenario: N/A

Exceptions: N/A

Relationships: N/A

Business Rules:

3.3. Software System Attributes

3.3.1. Usability

- Simple interface
- Easy to use

3.3.2. Maintainability

- Observe coding convention
- Comment
- Easy to develop new functions

3.3.3. Security

- Secret key is private, just only admin know to use.

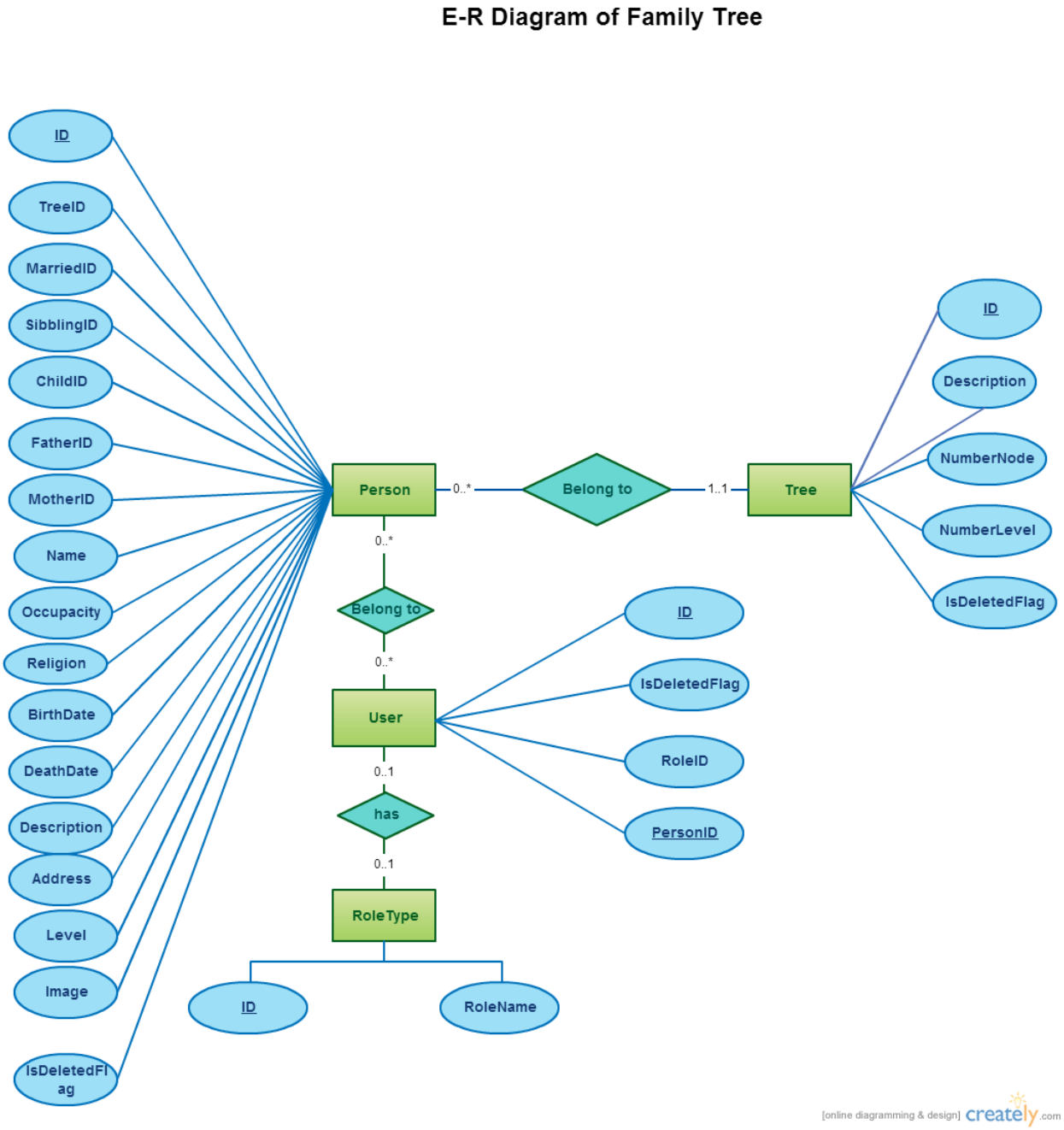
3.3.4. Portability

- Website can be deployed on every environment which support minimum requirement.

3.3.5. Performance

- **Load time:**
    + Website: Display in 5 seconds.
    + Comment and like function should affect immediately, without reloading the web page.
- **Compatibility:**
    + Display well on most of PC/laptop screen resolutions: 1024 x 768, 1280 x 800, and 1366 x 768.
    + The output must be compatible with all common browsers: Mozilla Firefox, Google Chrome andInternet Explorer.

3.4. Entity Relationship Diagram



Chapter 4: Software Design Description (SDD)

4.1. Design Overview

This document gives a detailed overview of Family Tree system's technical design and architecture design. It also includes the design of database model, the detailed design of common functions and business functions and overall design of user interface.

The technical design allows system builders (developers) to understand the actual hardware and software need to solve the problem.

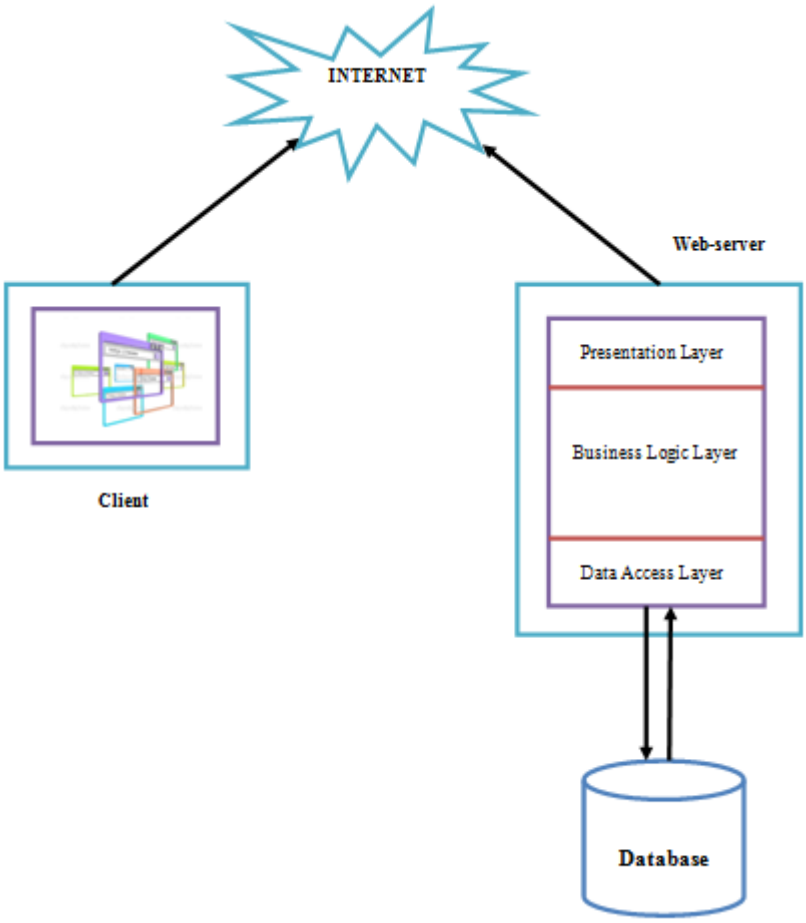
The architectural design describes the overall architecture of the system, and the architecture of each main component and subsystem. It will describe the patterns being used, the role of each component and the role of the system in the working environment.

The detailed design describes static and dynamic structure for each component and function. It includes class diagrams, class explanations, and sequence diagrams of the main use cases. The detailed design uses notations of **ASP.NET framework** as the framework for developing the system.

The database design describes the relationship between entities, and details of each entity. It uses notations of **Microsoft SQL Server** as it is the database server for developing the system.

The user interface design describes the layout of the system, and some design for the screens.

4.2. System Architectural Design



The system will use the Client-Server as the main architecture. It makes use of the Internet and common web protocols to deliver its services to the users.

The main entities of the above diagram will be described as following:

Database store: this is the central data storage for the system. It may be running on a separate server to the Application Server, or in the same machine.

Web-based Application: this is the server of the system running on Windows web server. It includes main components Web Application. Web Application is responsible for creating web pages. It is developed using **ASP.NET framework**. Web Services provides the system's API to the world and some parts of the application.

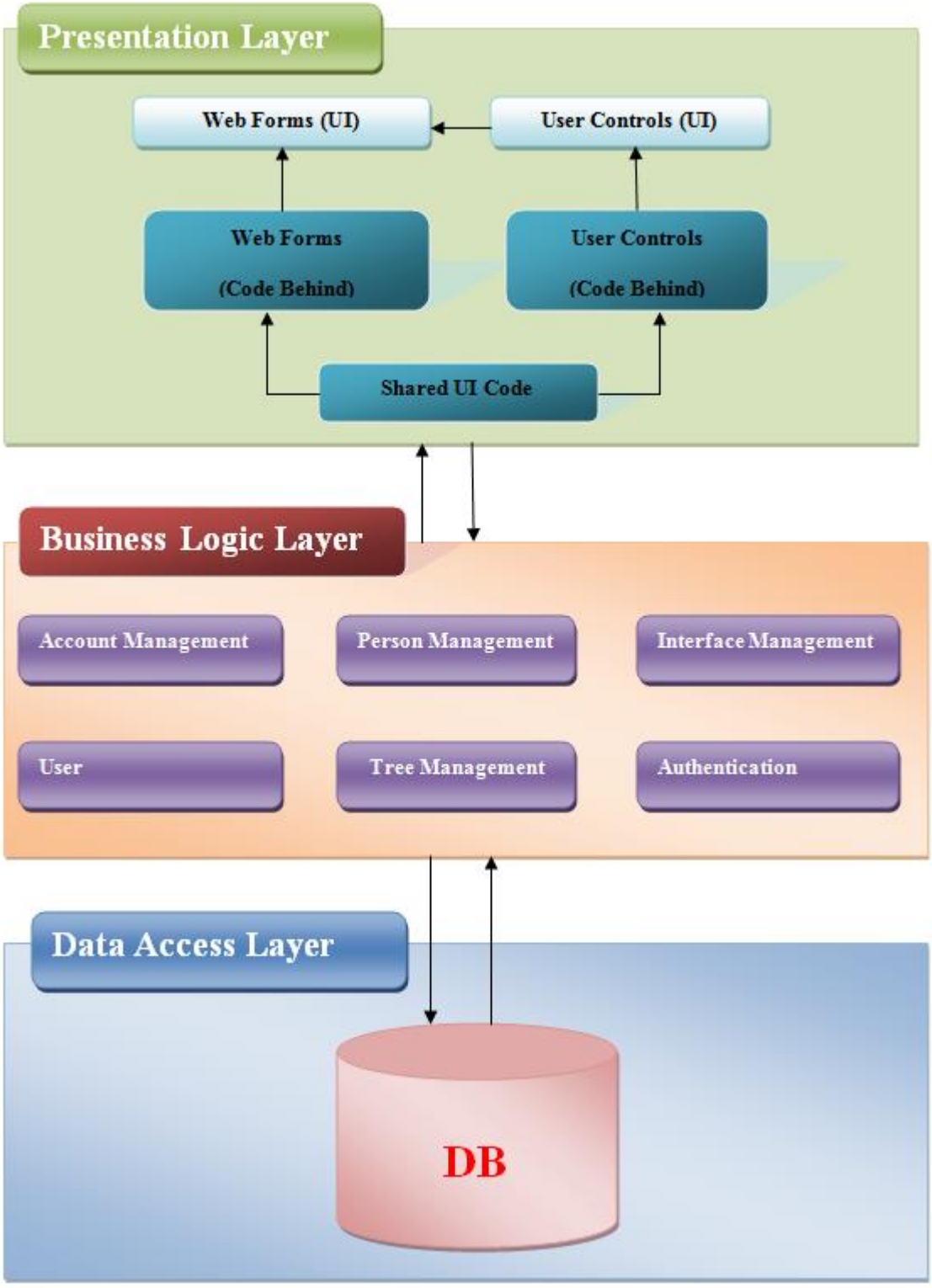
Service Providers: these are third party service providers that the application may need services from Facebook's API.

Clients: these include internet browsers running on personal computers, or mobile applications targeting mobile devices. The web browsers will interact with the web application HTTP protocols.

4.2.1. System Architectural

The system will use the 3-tier architectural as the main architecture. The main entities of the this architectural will be described as following:

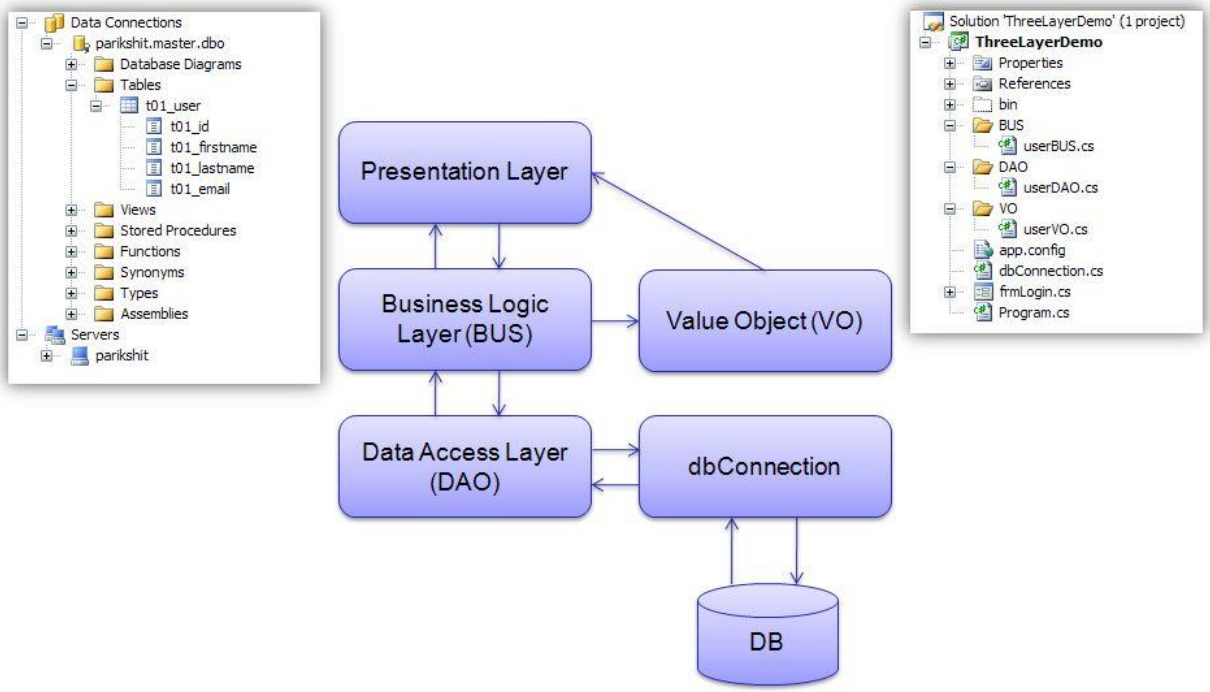
- **Presentation tier**: Web Form(UI), Web Form(Code Behind), User Controls(UI), User Controls(Code behind) and Shared UI Code
- **Business tier**: Business Logic layer, Value Object, Data Access Layer
- **Data tier**: Stored Process, XLST, …



- Data Tier is basically the server which stores all the application's data. Data tier contents Database Tables, XML Files and other means of storing Application Data.
- Business Tier is mainly working as the bridge between Data Tier and Presentation Tier. All the Data passes through the Business Tier before passing to the presentation Tier. Business Tier is the sum of Business Logic Layer, Data Access Layer and Value Object and other components used to add business logic.

- Presentation Tier is the tier in which the users interact with an application. Presentation Tier contents Shared UI code, Code Behind and Designers used to represent information to user.

4.2.1.1. 3 Layer Windows Application in C#.NET



dbConnection

This class is mainly used to do the database activity like Select, Update and Delete query to database. It also checks if the database connection is open or not. If database connection is not open, then it opens the connection and performs the database query. The database results are to be received and being passing in Data Table in this class.

This class takes the database setting from the app.config file so it's really flexible to manage the database settings.

Database Access Layer

Database Access Layer (DAO) builds the query based on received parameters from the Business Logic Layer and passes it the dbConnection class for execution. And simple return results from the dbConnection class to Business Logic Layer.

Value Object

Value Object is nothing more but a class with the contents GET and SET methods. It's mainly used to pass Data from one class to another. It's directly connected with Business Logic Layer and Presentation Layer. As you can see in the diagram object values are being SET in Business Logic Layer and GET from Presentation Layer.

Business Logic Layer

Business Logic Layer (BUS) works as a bridge between Presentation Layer and DAO. All the user values received from the presentation layer are being passed to BUS. The results received from the DAO are in row data in Data Table format but in BUS it's converting into Value

Objects (VO). Business Logic Layer (BUS) is the most important class in the whole architecture because it mainly contains all the business logic of the program. Whenever a user wants to update the business logic of the program only need to update this class.

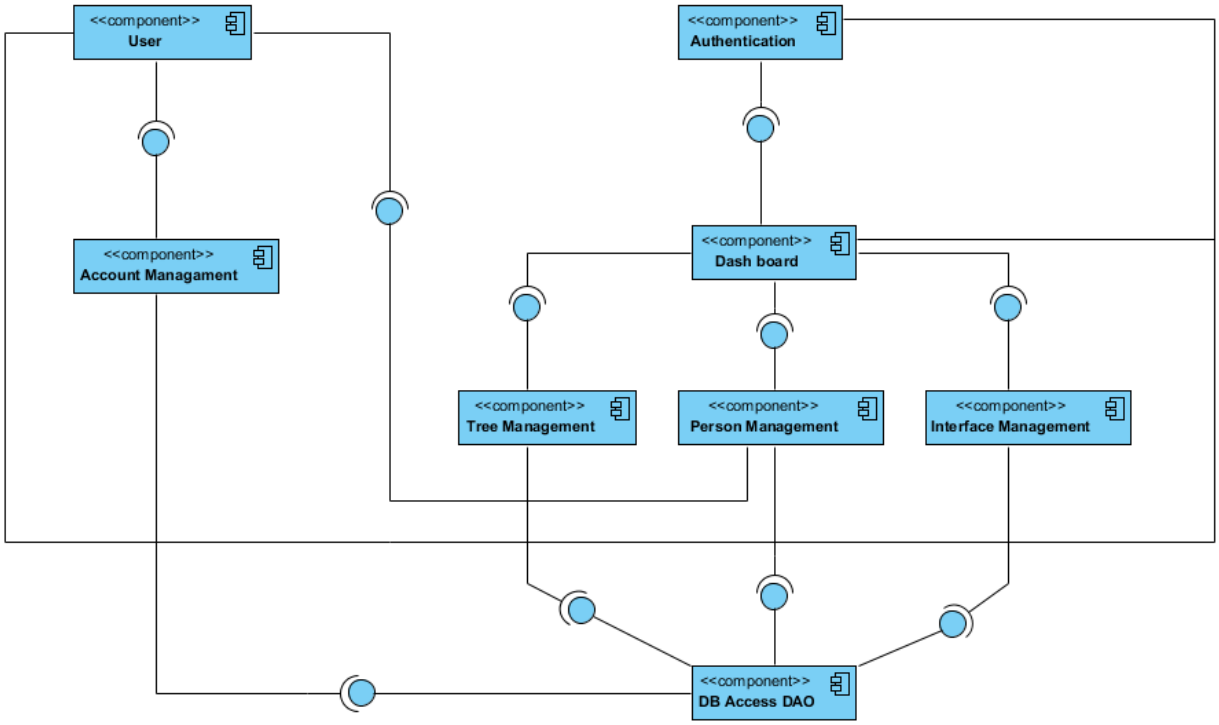
Presentation Layer

Presentation Layer is the only layer which is directly connected with the user. So in this matter, it's also a really important layer for marketing purposes. Presentation Layer is mainly used for getting user data and then passing it to Business Logic Layer for further procedure, and when data is received in Value Object then it's responsible to represent value object in the appropriate form which user can understand.

4.2.2. Alternative Designs

4.2.3. System Interface

4.3. Component Diagrams



4.4. Detailed Description of Components

4.4.1. CRC Cards (Class-Responsibility-Collaborators)

Database	
Responsibilities	**Collaborators**
Connect Database	AccountManagement
Close Connection	TreeManagement
Create Command	InterfaceManagement
Execute	PersonManagement
ExecuteReader	

Person	
Responsibilities	**Collaborators**
Create Object Person	PersonManagement
	Tree
	Child
	Sibling
	Married

Tree	
Responsibilities	**Collaborators**
Create Object Tree	TreeManagement
	Person

User	
Responsibilities	**Collaborators**
Create Object User	Role
	AccountManagement

Role	
Responsibilities	**Collaborators**
Create Object Role	Database
	User

TreeManagement	
Responsibilities	**Collaborators**
EditTree()	Database
DeleteTree()	Tree
Save()	User
Search()	
SearchBetweenManyFamilies()	

PersonManagement	
Responsibilities	**Collaborators**
AddPerson()	Database
EditPerson()	Person
DeletePerson()	

InterfaceManagement	
Responsibilities	**Collaborators**
DisplayTree()	Database
ExpandTree ()	Person
ColapseTree()	Tree
DisplayPage()	Role
	UserHomePage
	SystemAdminPage
	FamilyAdminPage
	PersonManagement
	TreeManagement
	AccountManagement

AccountManagement	
Responsibilities	**Collaborators**
AcceptUser()	Database
EditUser()	User
DeleteUser()	Role

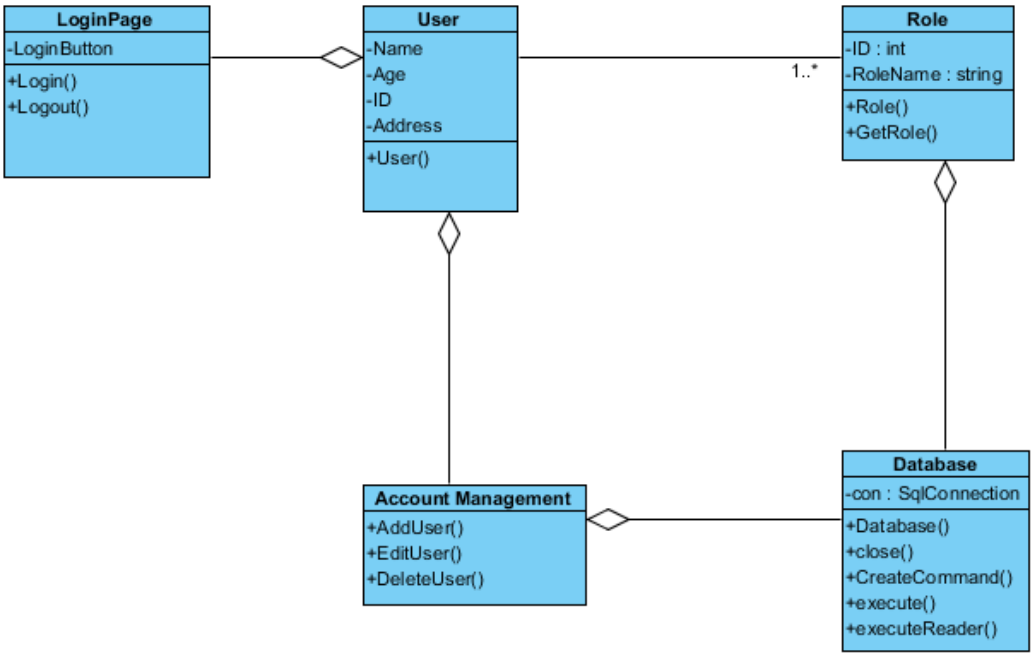
UserHomePage	
Responsibilities	**Collaborators**
Page_load()	Database
	InterfaceManagement

SystemAdminPage	
Responsibilities	**Collaborators**
Page_load()	Database
	InterfaceManagement

FamilyAdminPage	
Responsibilities	**Collaborators**
Page_load()	Database
	InterfaceManagement

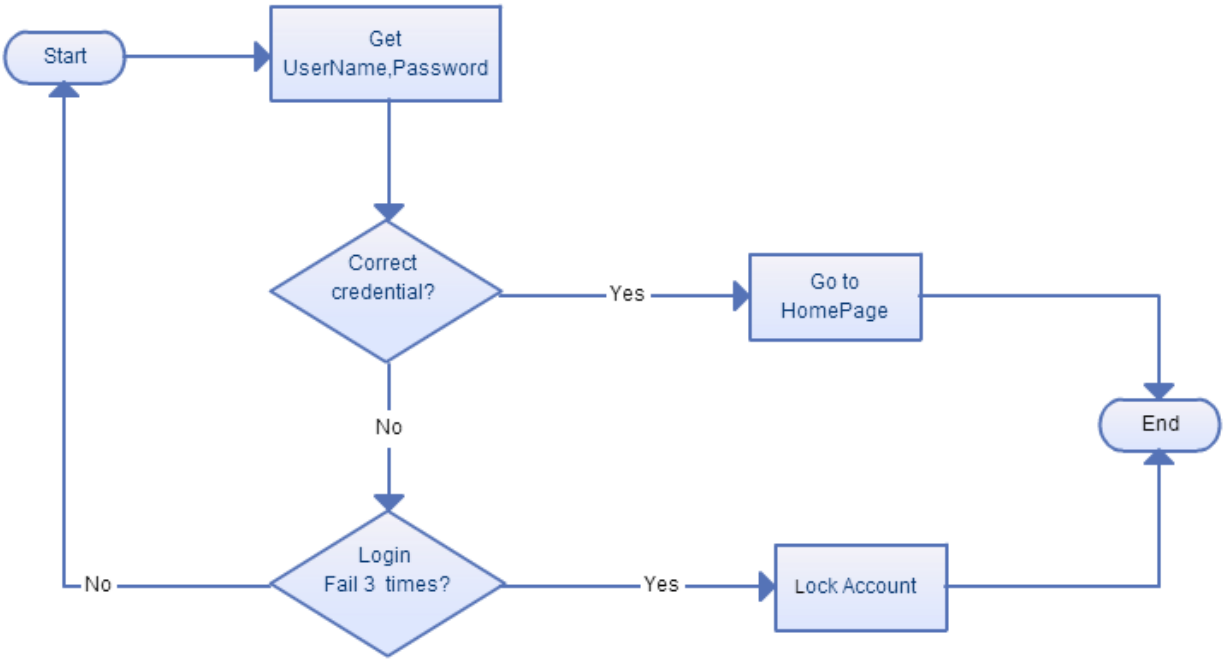
4.4.1.1. Component 1: Authentication

4.4.1.1.1. Class Diagram

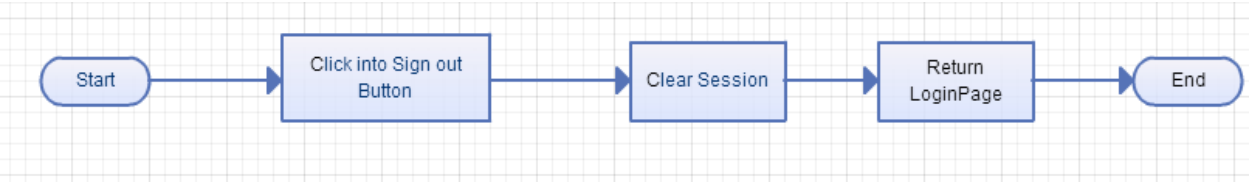


4.4.3.1.2. Flow-Chart

a. For method Login() of class Authentication

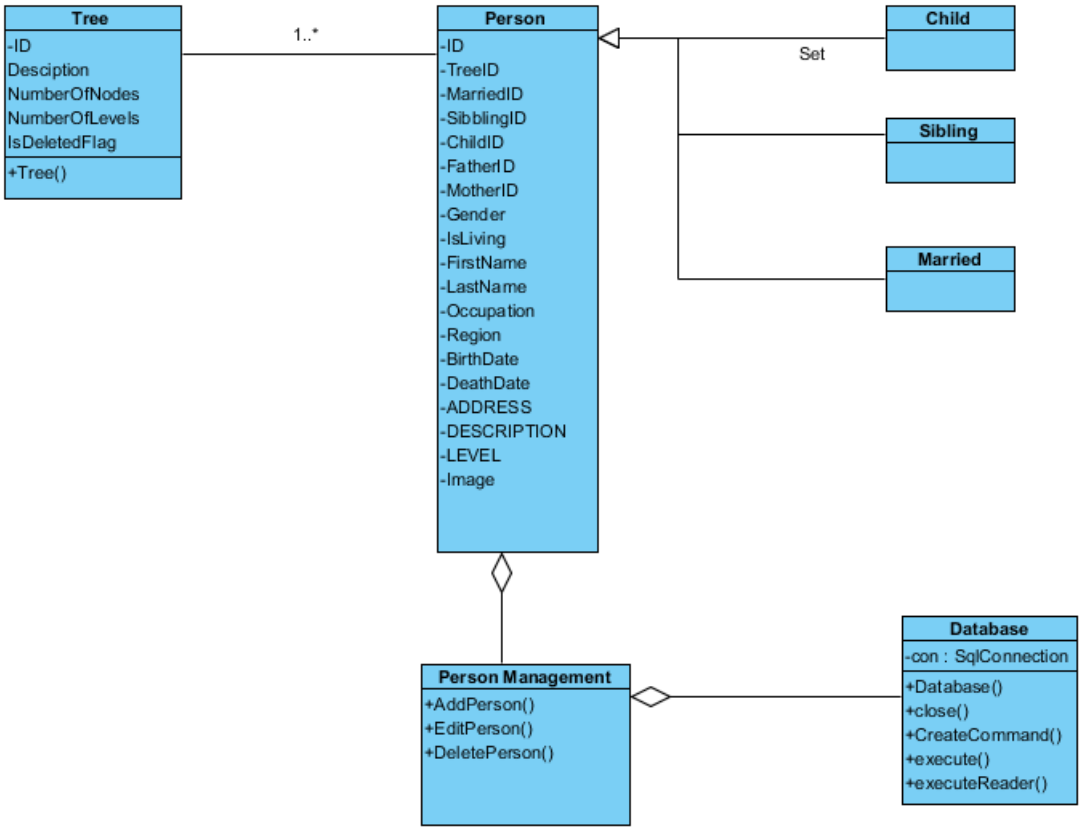


b. For method Logout() of class Authentication



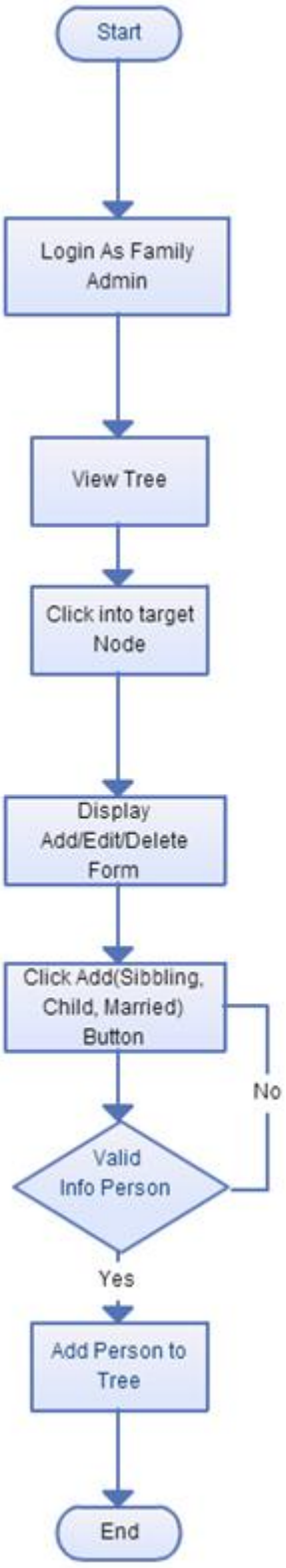
4.4.1.2. Component 2: Person Management

4.4.1.2.1. Class Diagram

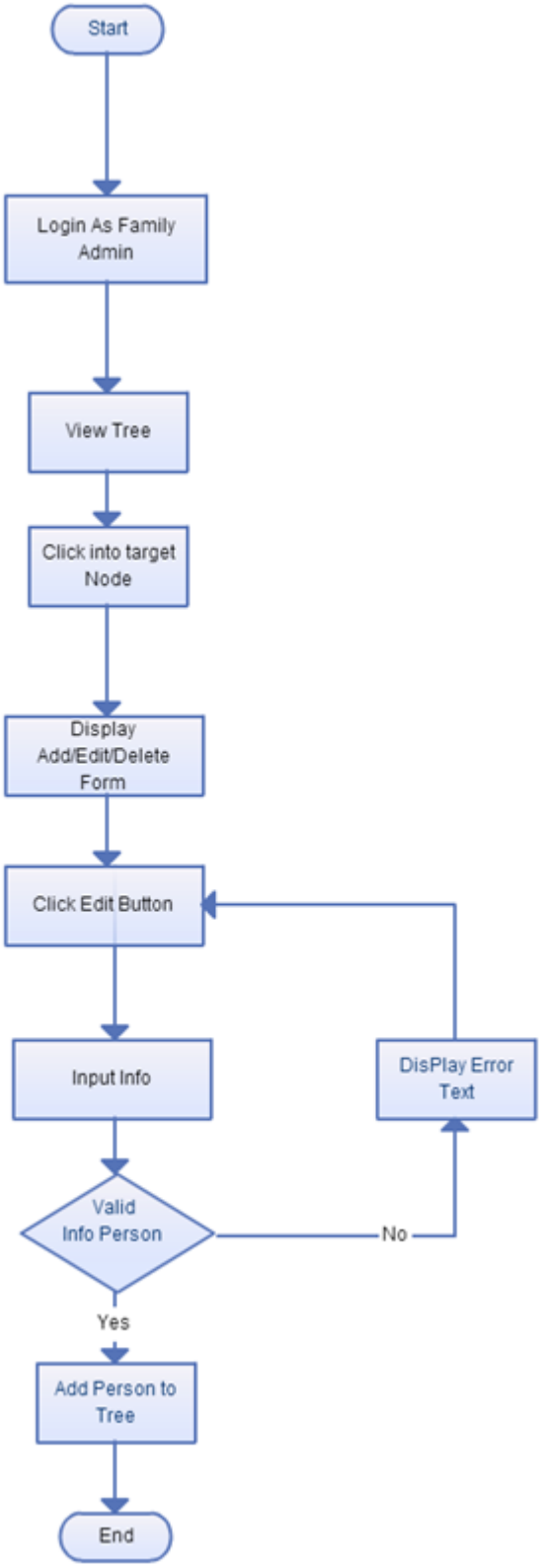


4.4.1.2.2. Flow-Chart

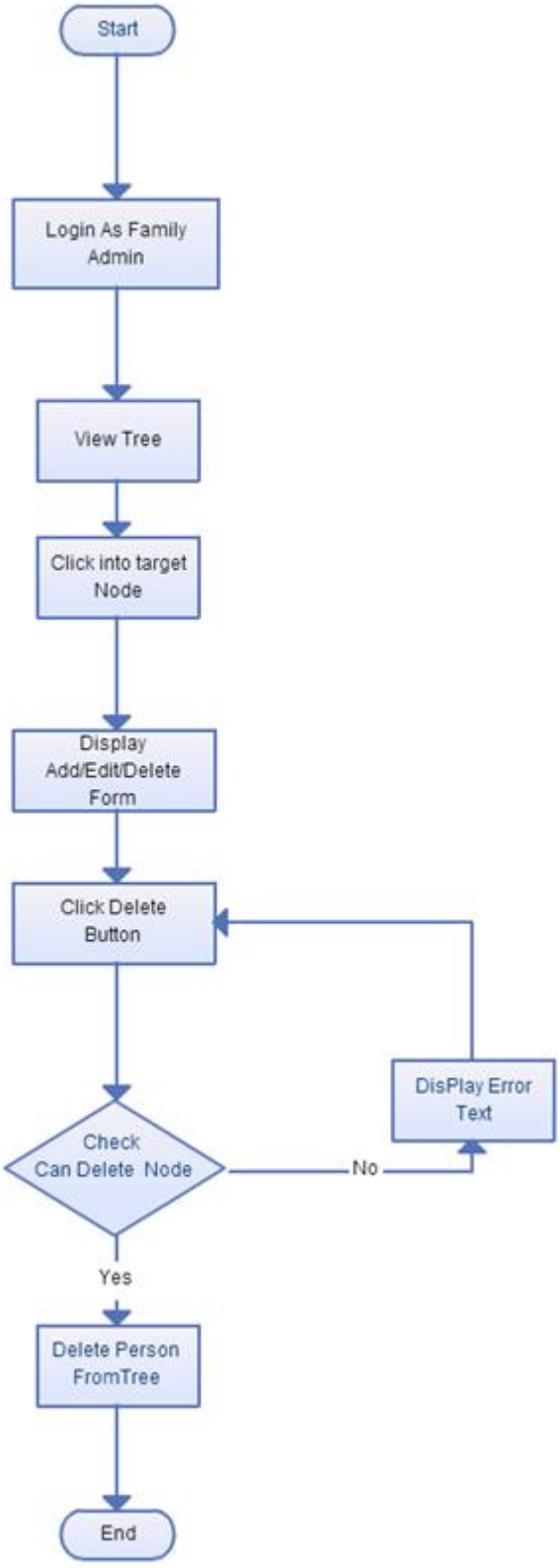
a. For method AddPerson() of class PersonManagement



b. For method EditPerson() of class PersonManagement

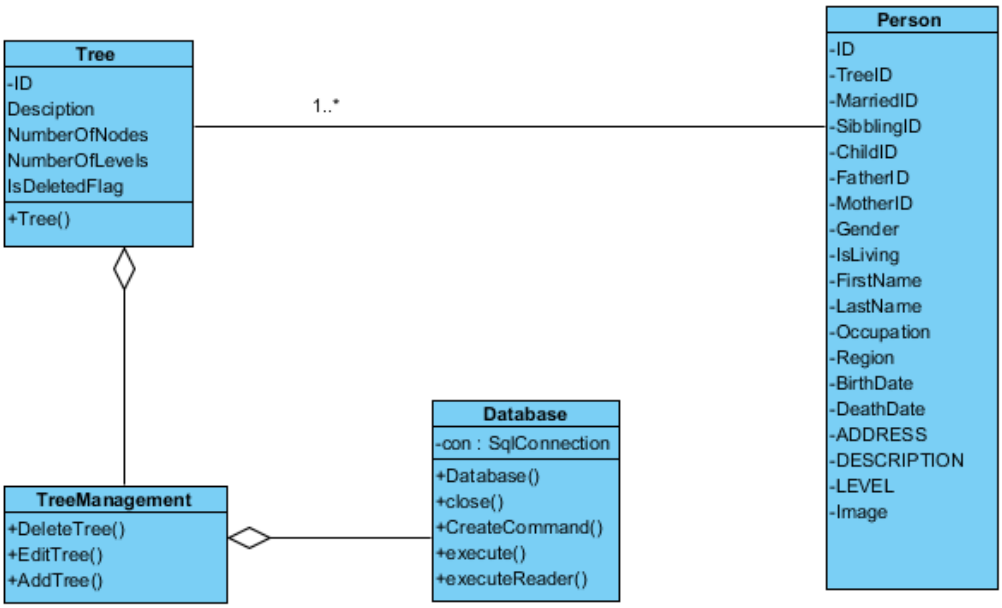


c. For method DeletePerson() of class PersonManagement


Start
Login As Family Admin
View Tree
Click into target Node
Display Add/Edit/Delete Form
Click Delete Button
Check Can Delete Node
No
DisPlay Error Text
Yes
Delete Person FromTree
End

4.4.1.3. Component-3: Tree Management

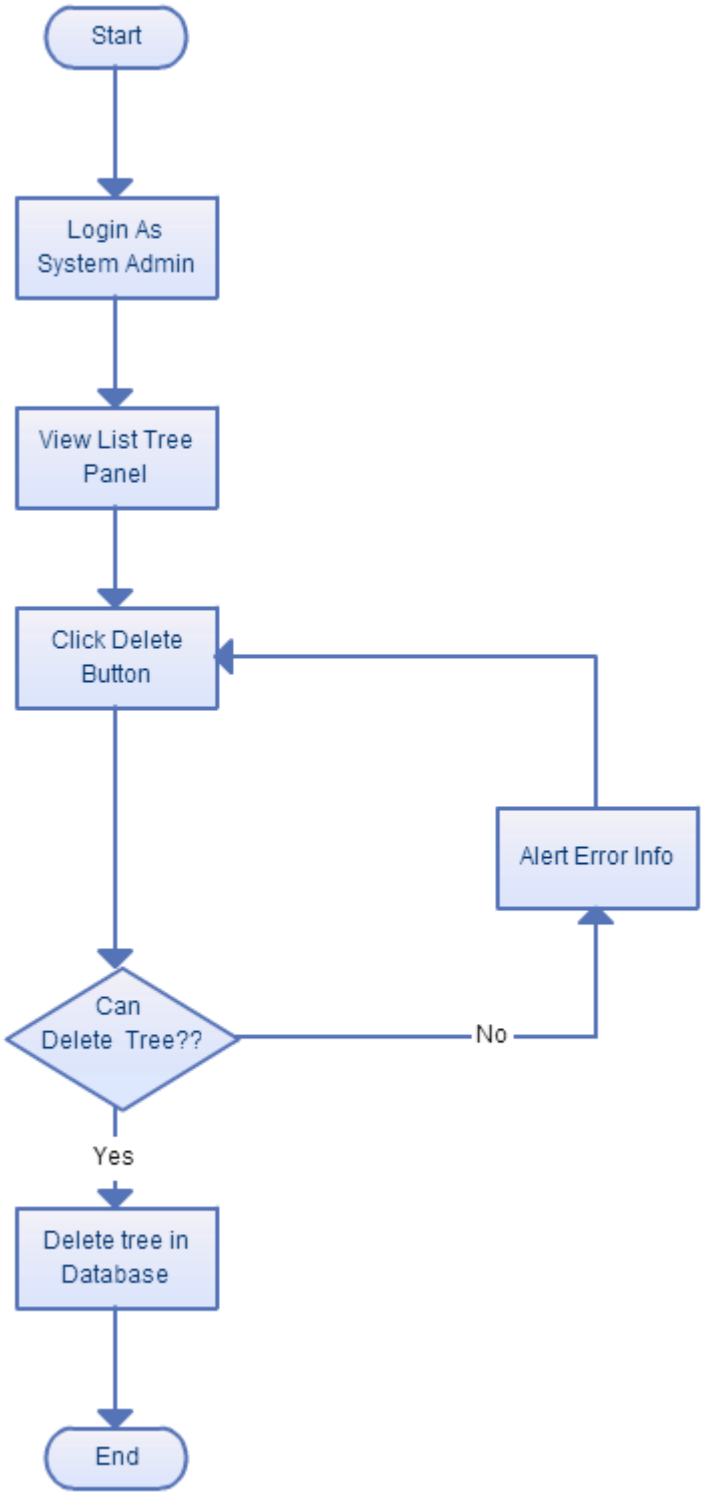
4.4.1.3.1. Class Diagram



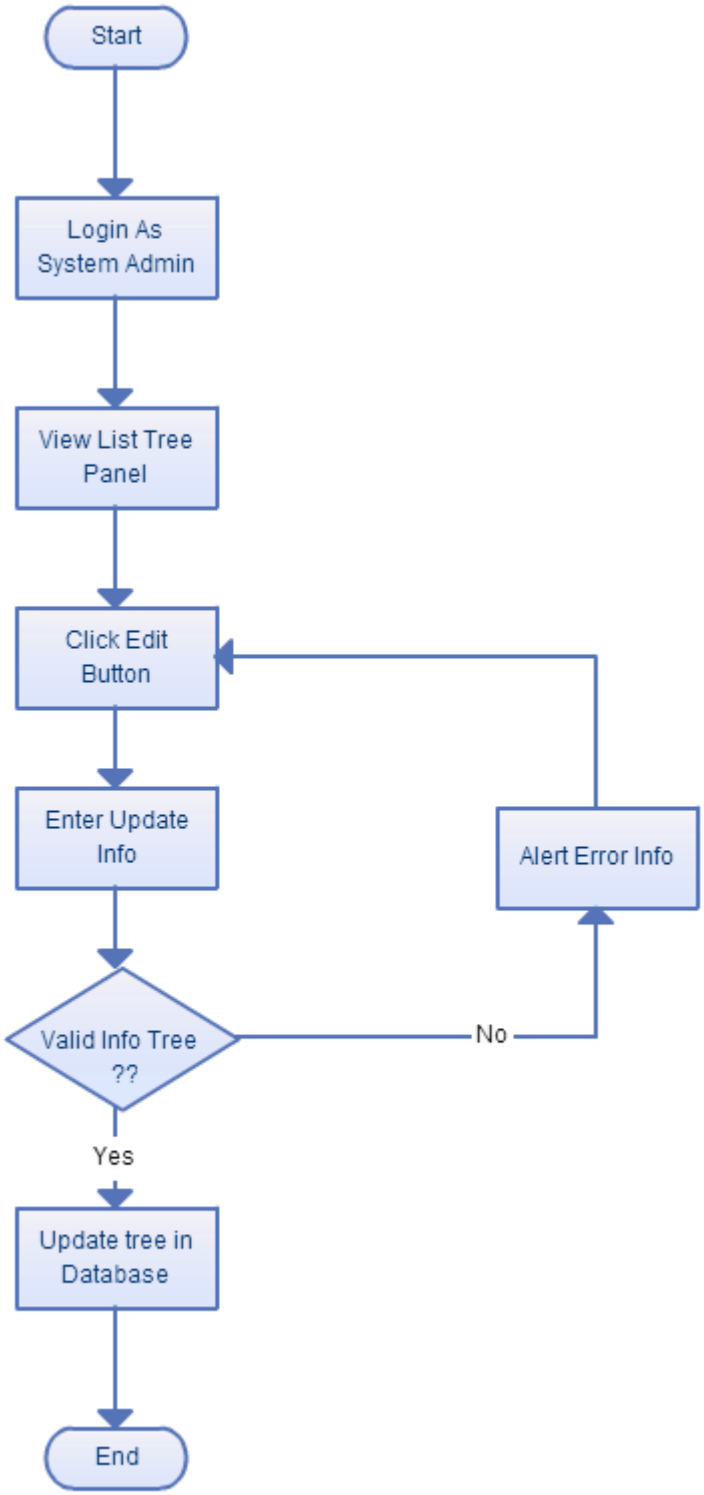
4.4.1.3.2. Class Diagram Explanation

4.4.1.3.3. Flow-Chart

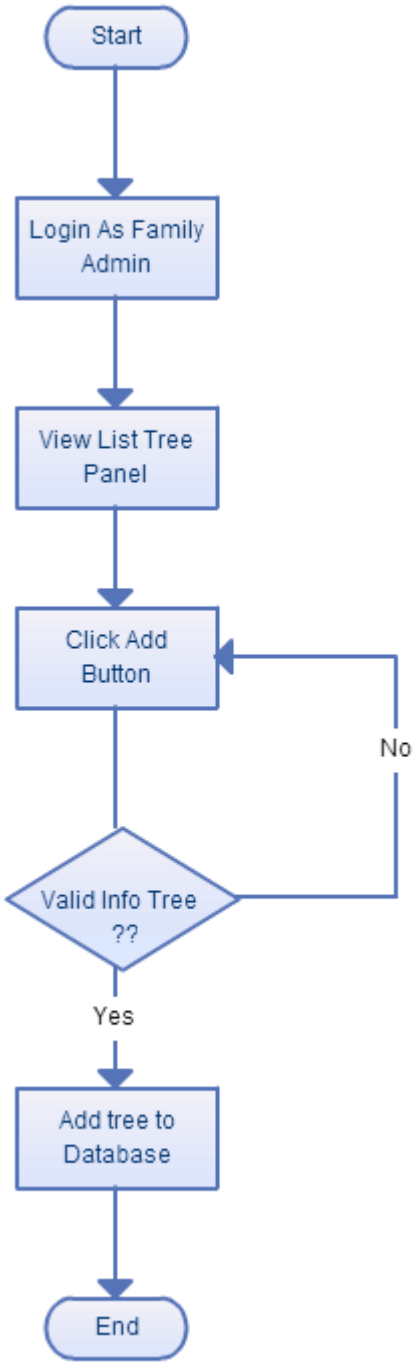
a. For method DeleteTree() of class TreeManagement



b. For method EditTree() of class TreeManagement

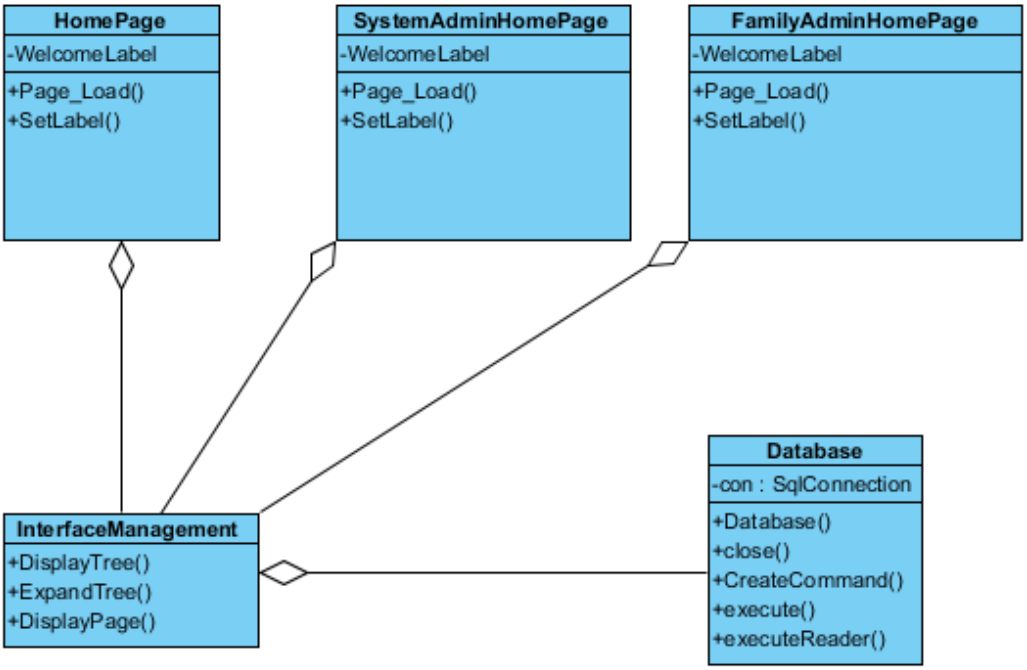


c. For method AddTree() of class TreeManagement


Start
Login As Family Admin
View List Tree Panel
Click Add Button
No
Valid Info Tree ??
Yes
Add tree to Database
End

4.4.1.4. Component-4: Interface Management

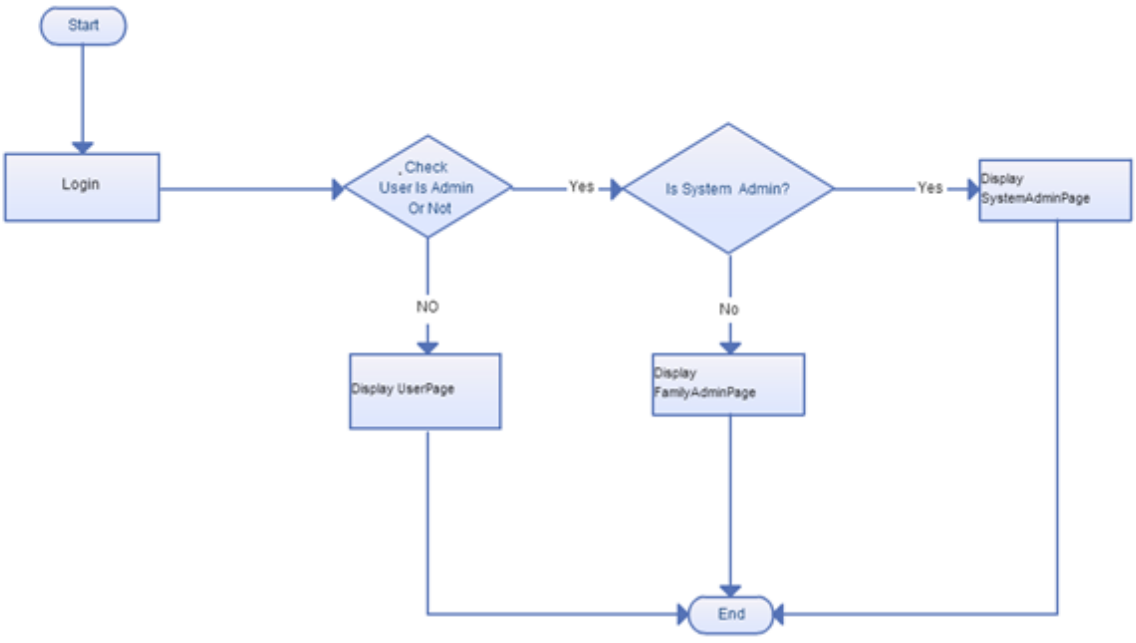
4.4.1.4.1. Class Diagram



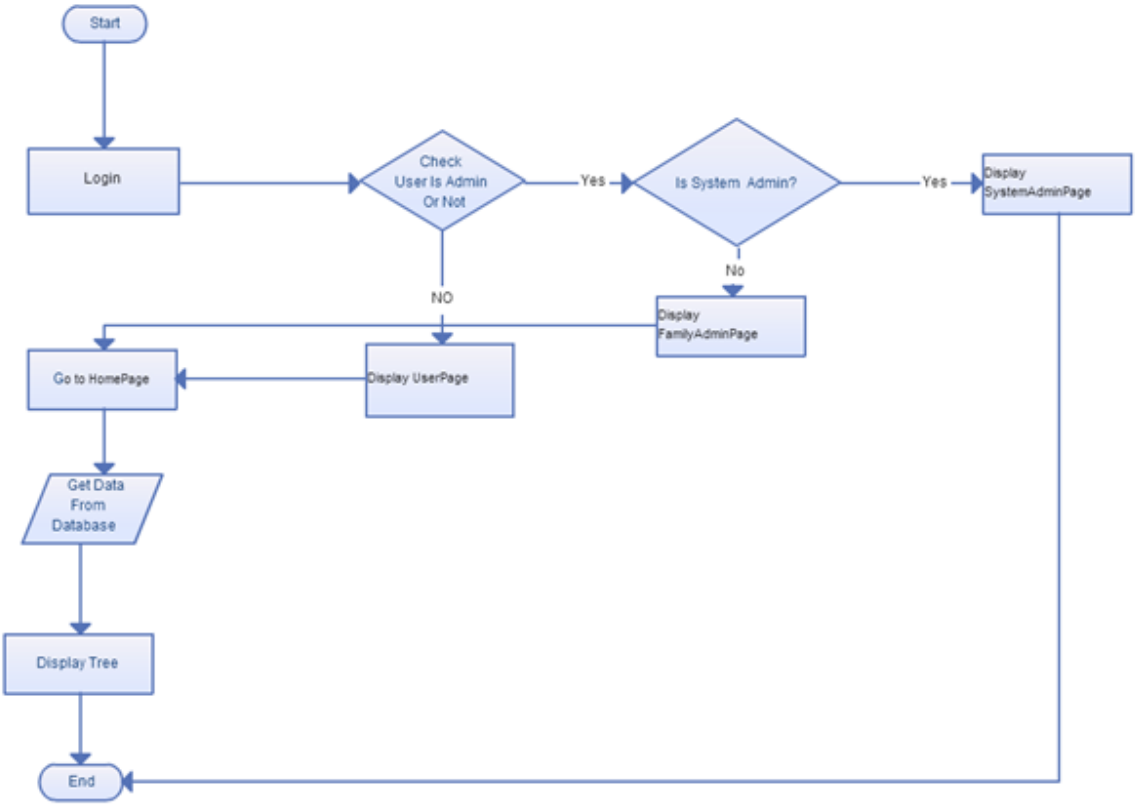
4.4.1.4.2. Class Diagram Explanation

4.4.1.4.3. Flow-Chart

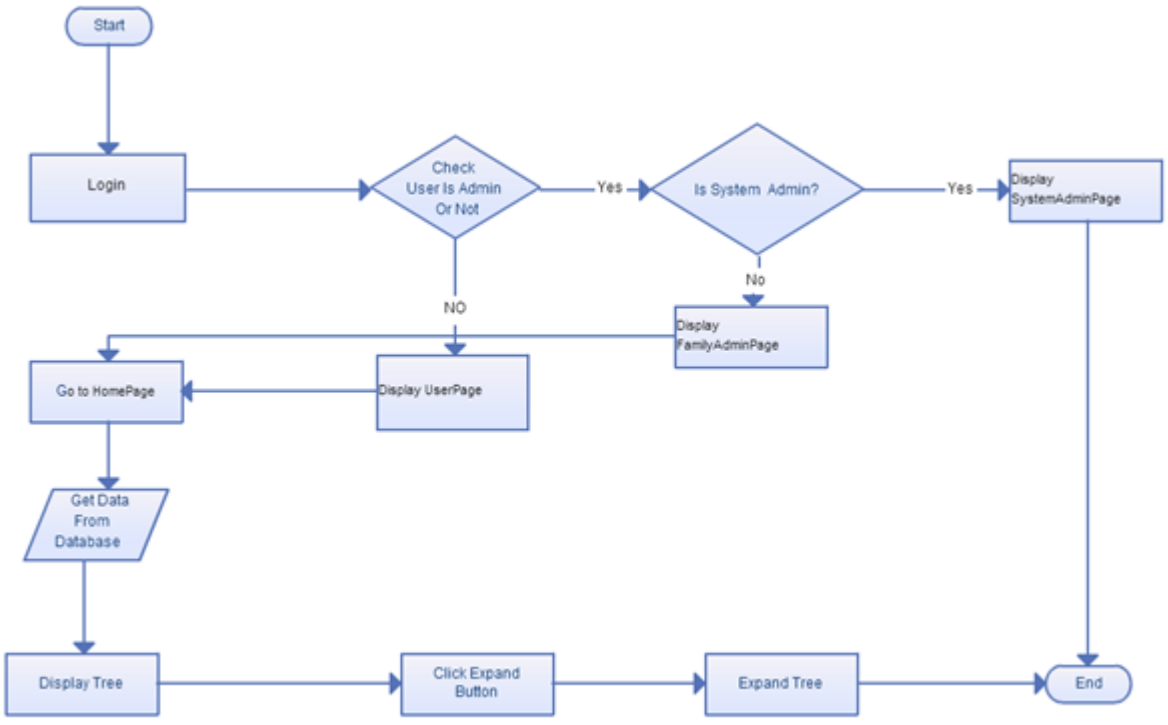
a. For method DisplayPage() of class InterfaceManagement



b. For method DisplayTree() of class InterfaceManagement

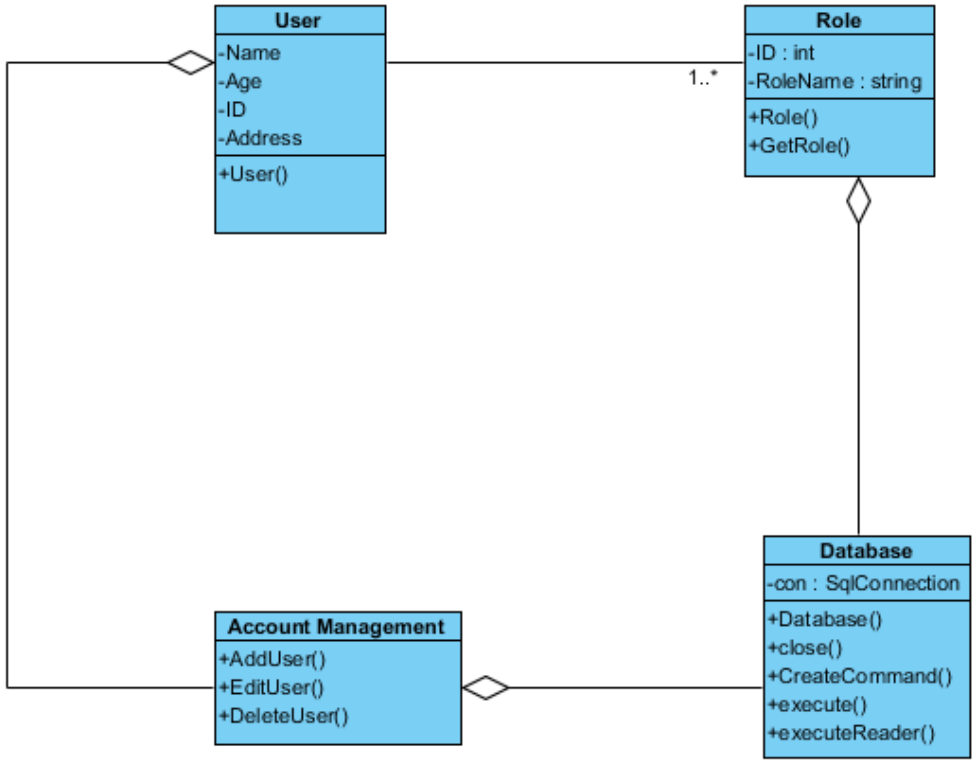


c. For method ExpandTree() of class InterfaceManagement



4.4.1.5. Component-5: Account Management

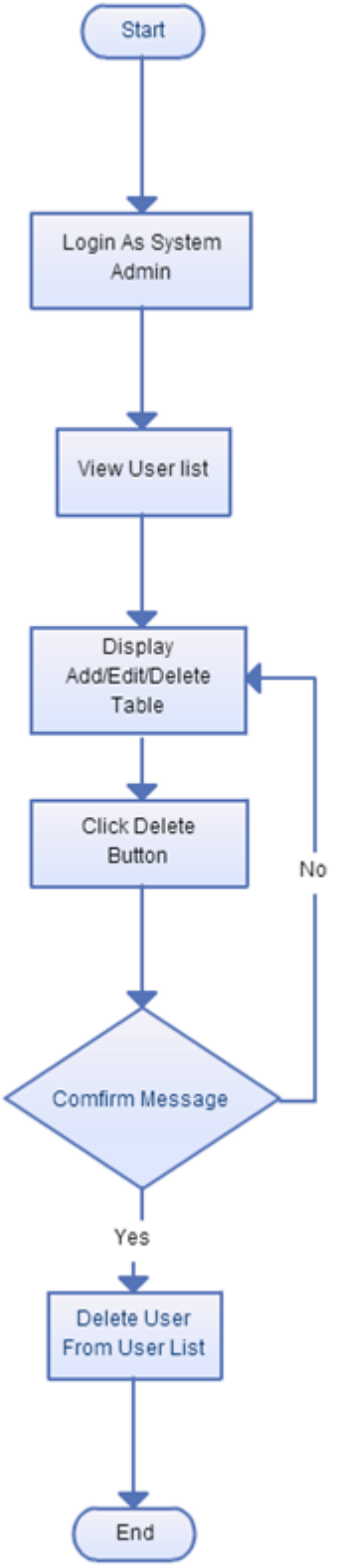
4.4.1.5.1. Class Diagram



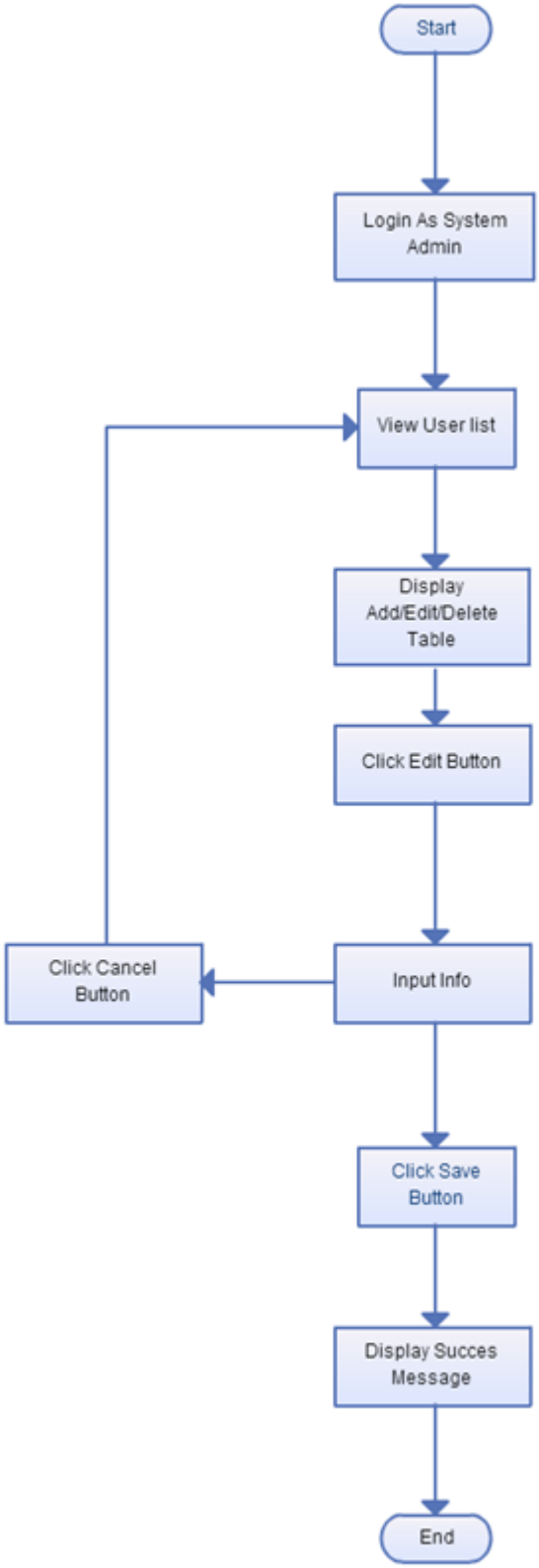
4.4.1.5.2. Class Diagram Explanation

4.4.1.5.3. Flow-Chart

a. For method DeleteUser() of class AccountManagement

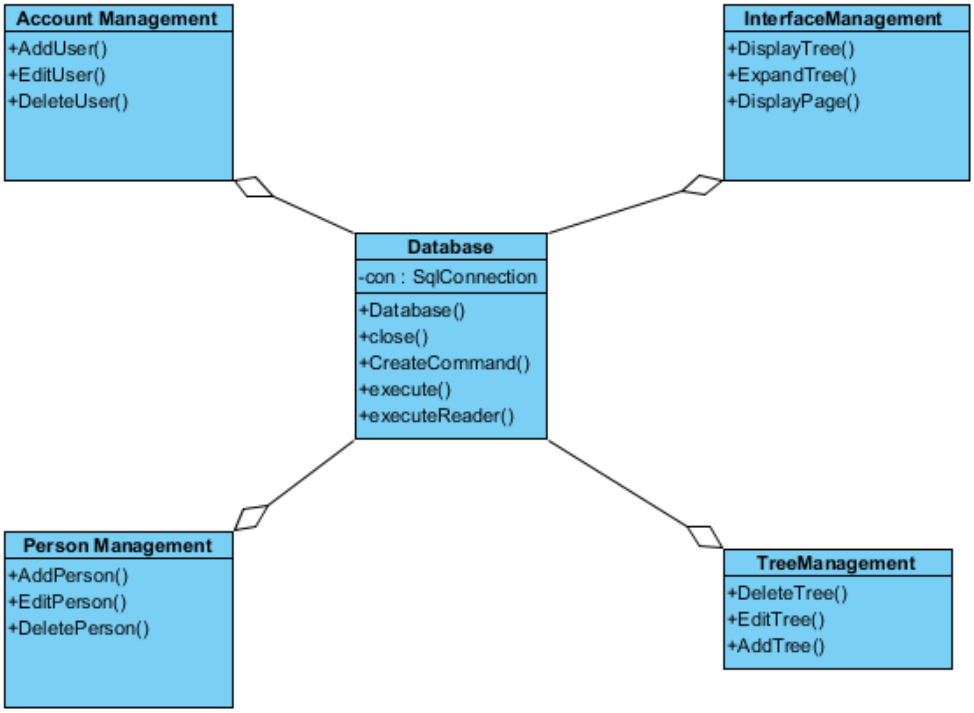


b. For method EditUser() of class AccountManagement



4.4.1.6. Component-6: Database Access

4.4.1.6.1. Class Diagram

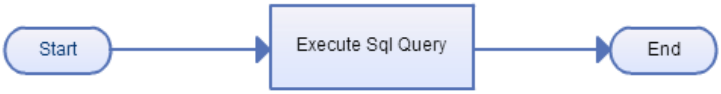


4.4.1.6.2. Flow-Chart

Database()

CreateCommand()

Execute()

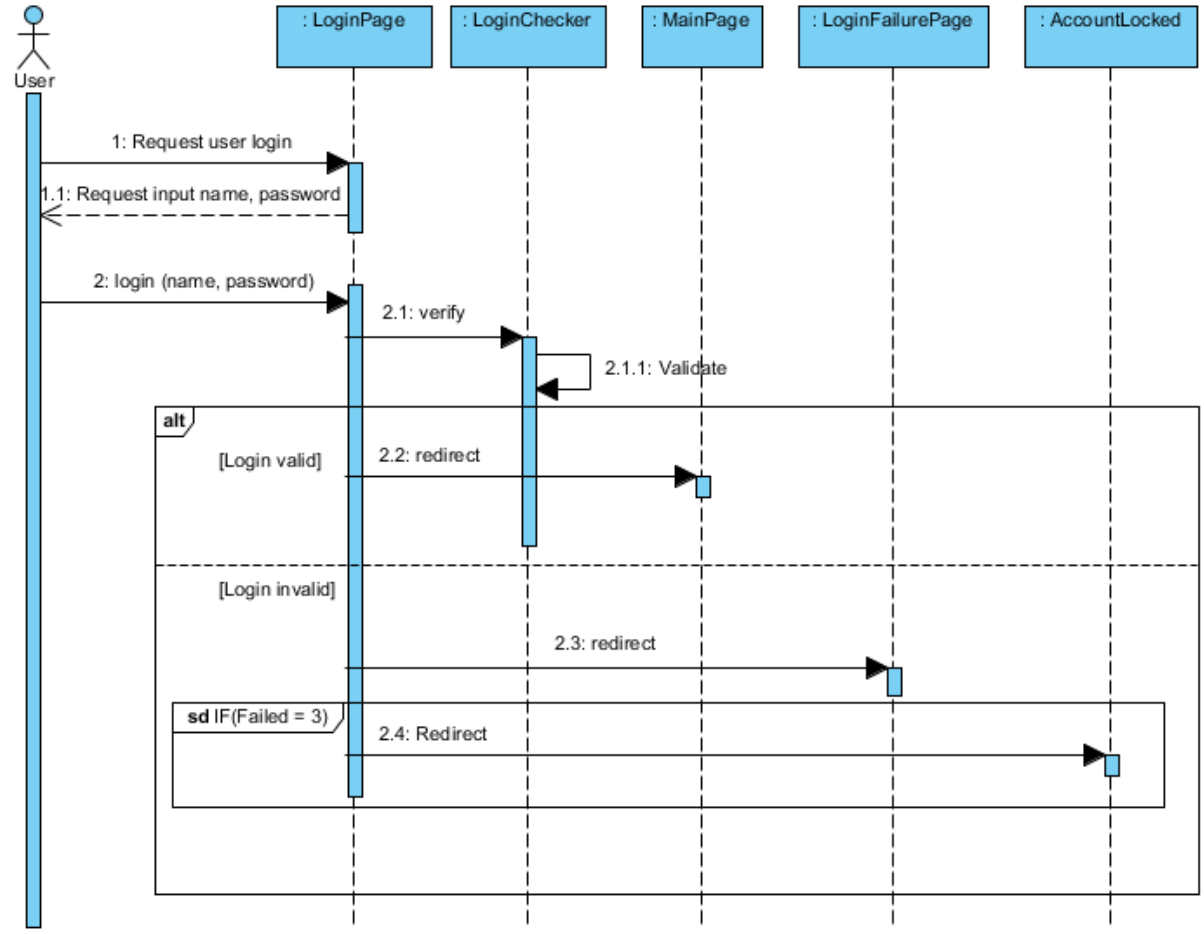


ExecuteReader()

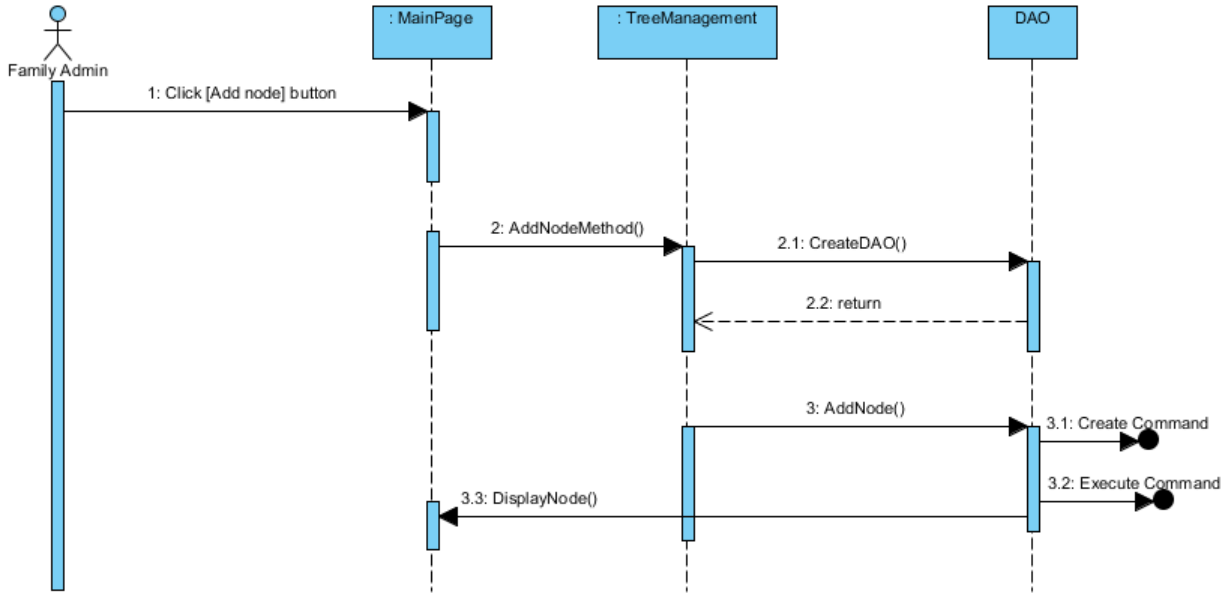
Close()

4.5. Sequence Diagram

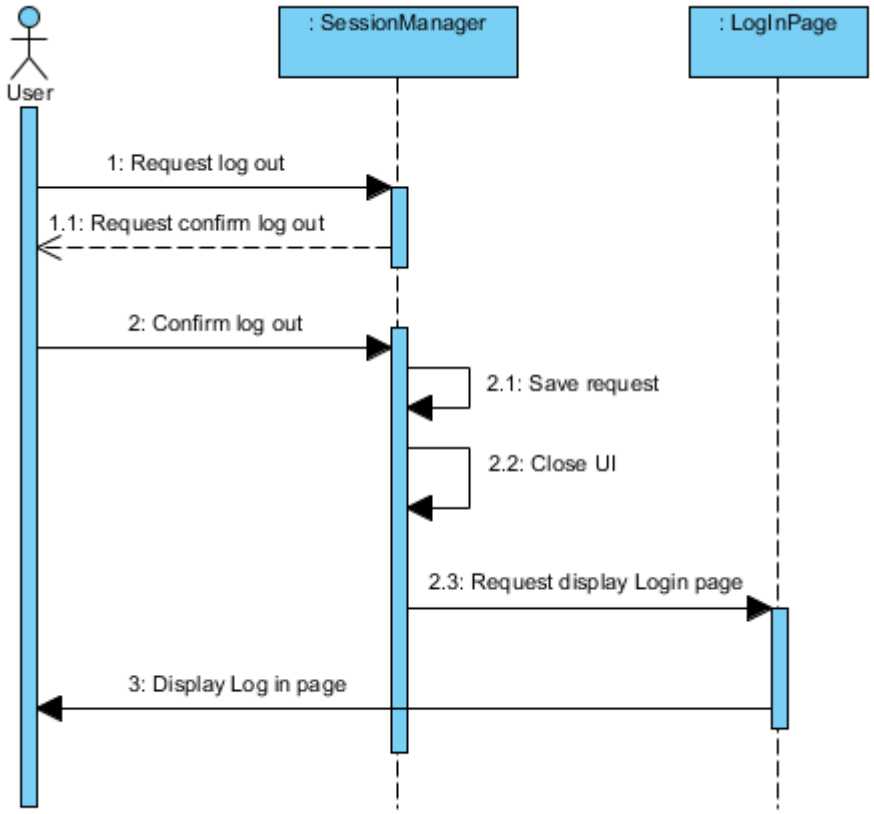
4.5.1. For use case "*Login*"



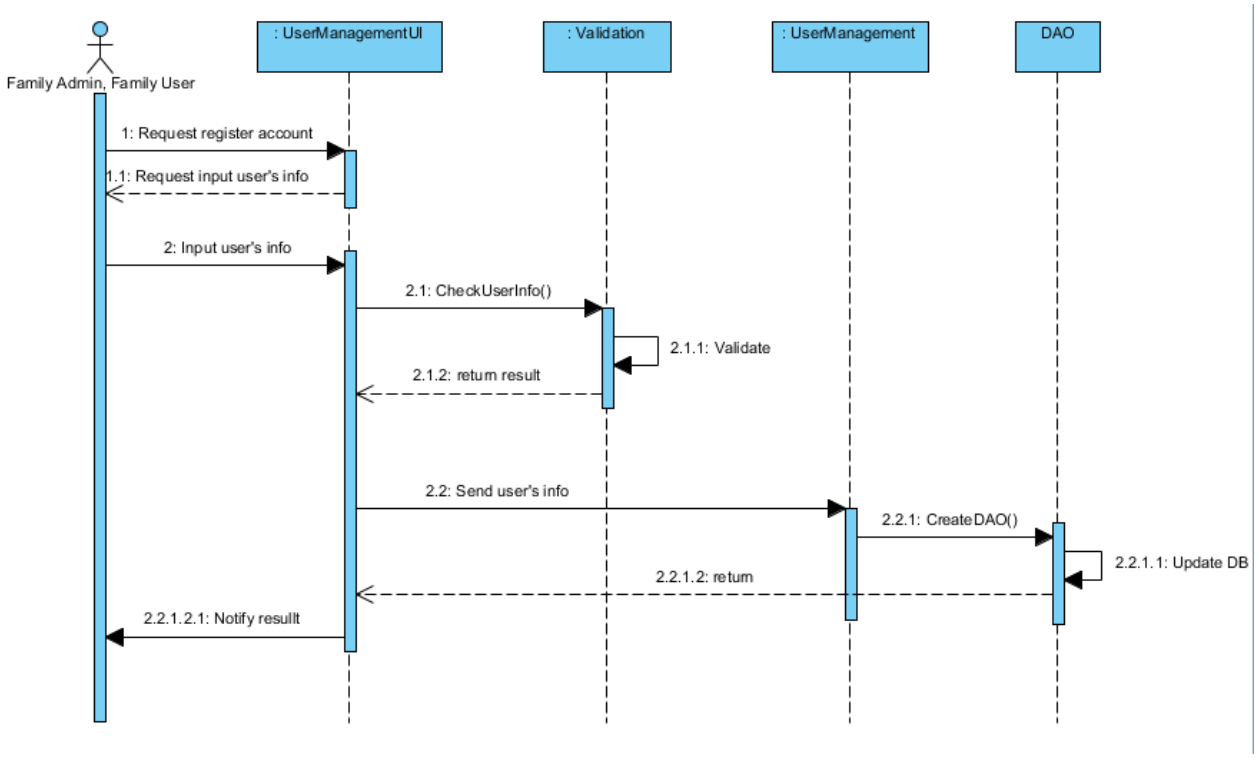
4.5.2. For use case "*Create Node*"



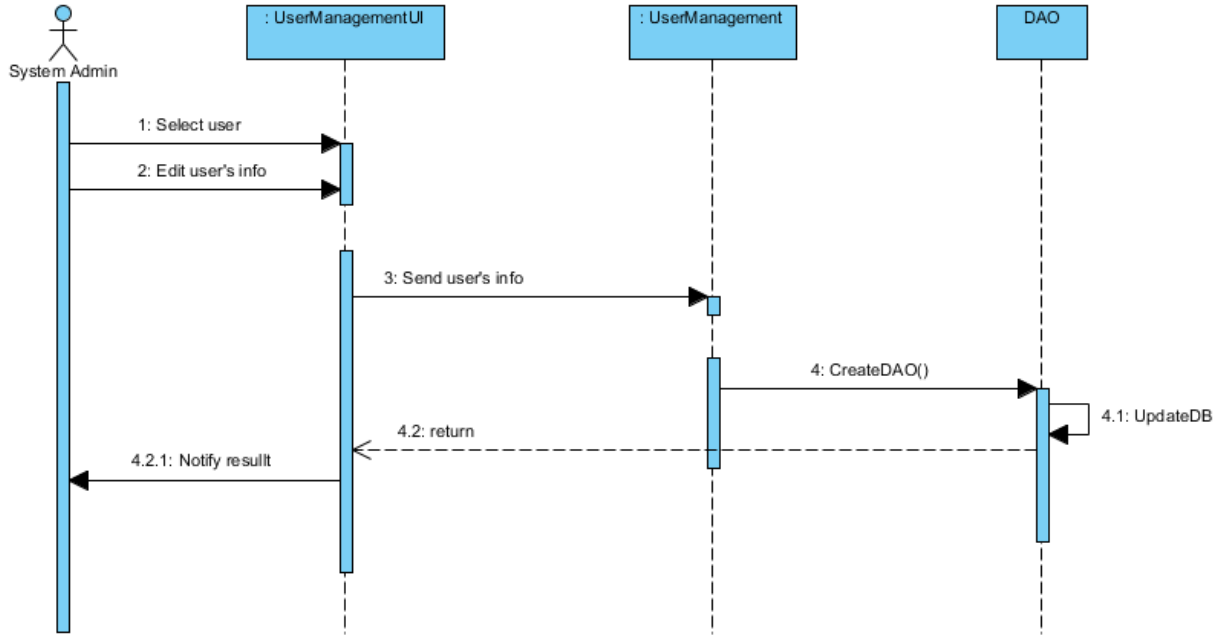
4.5.3. For use case "*Log out*"



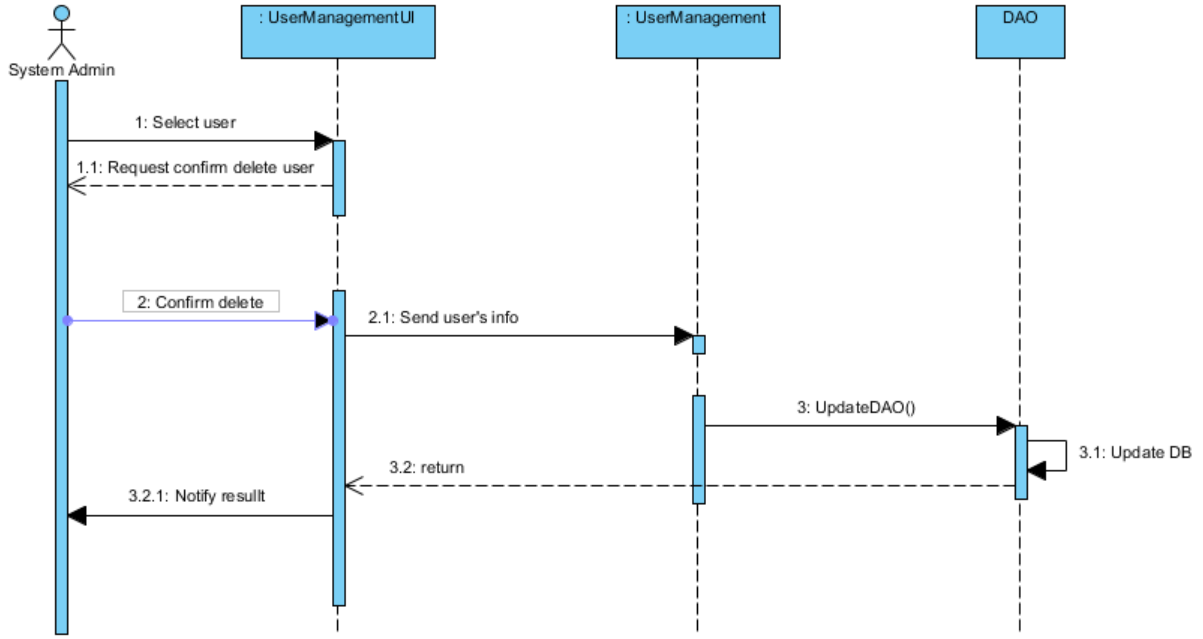
4.5.4. For use case "*Create new user account*"



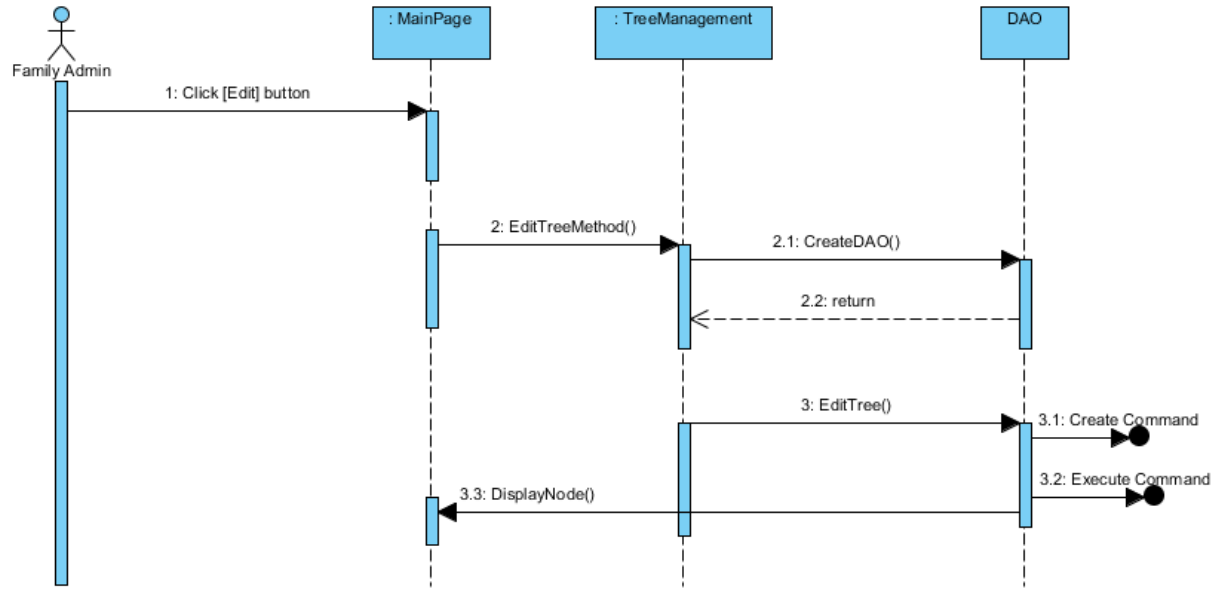
4.5.5. For use case "*Edit user account*"



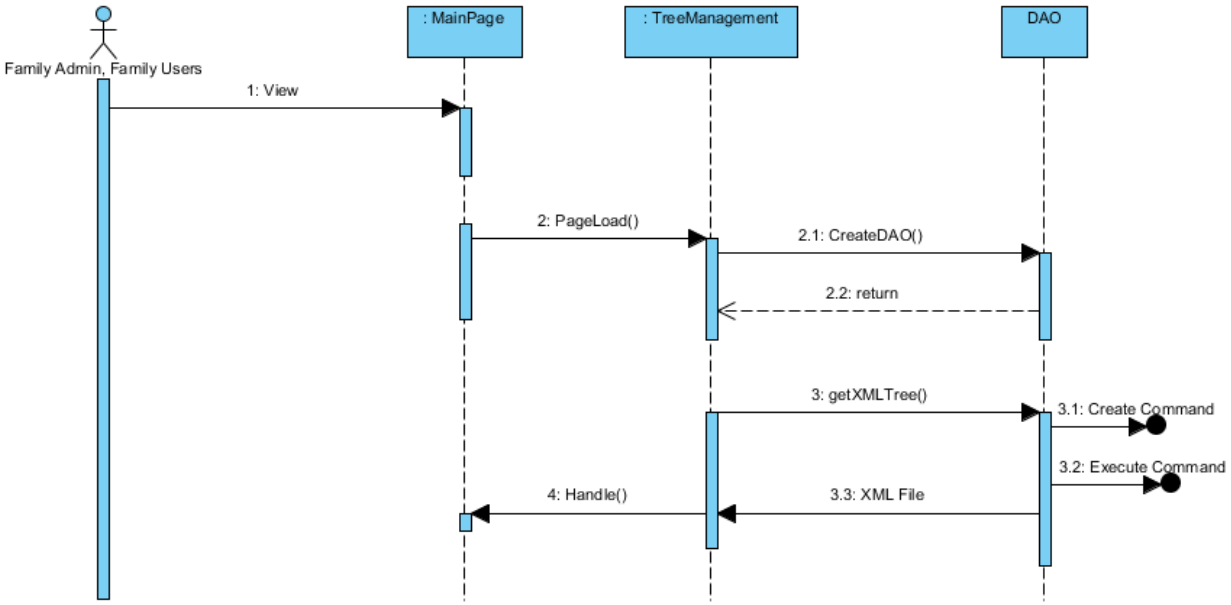
4.5.6. For use case "*Delete user account*"



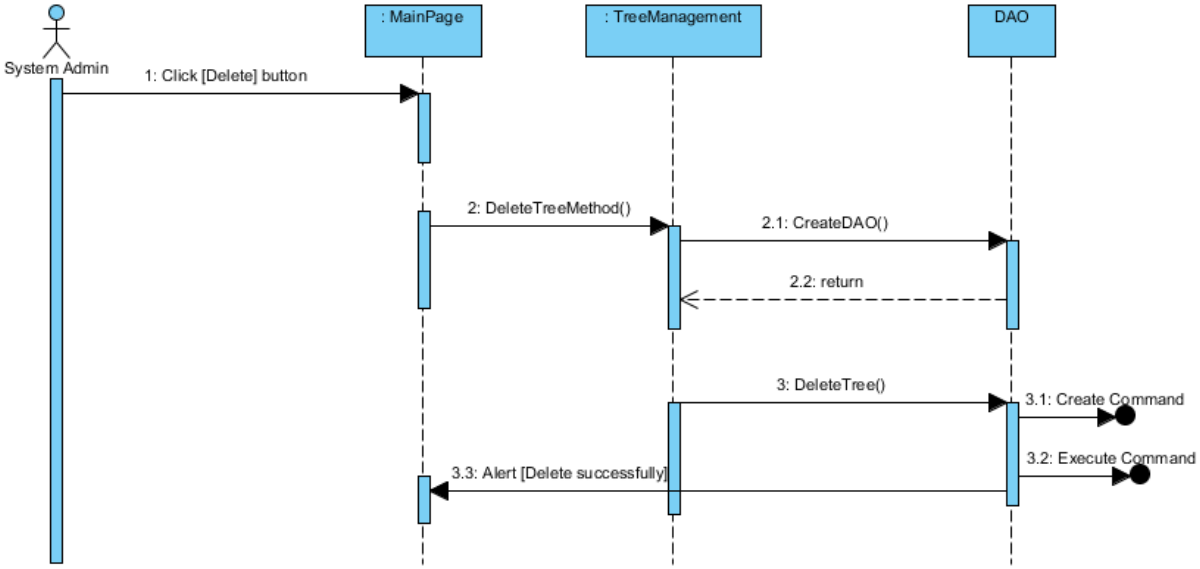
4.5.7. For use case "*Edit Node*"



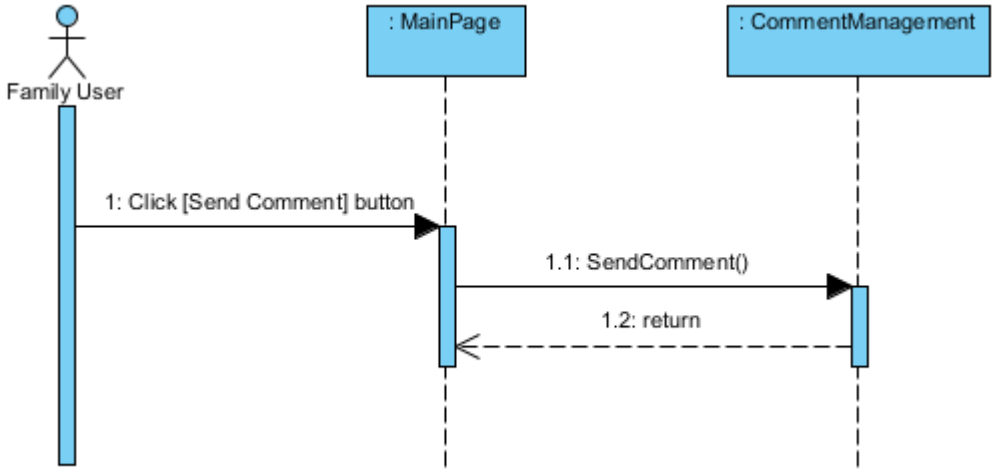
4.5.8. For use case "*View Tree*"



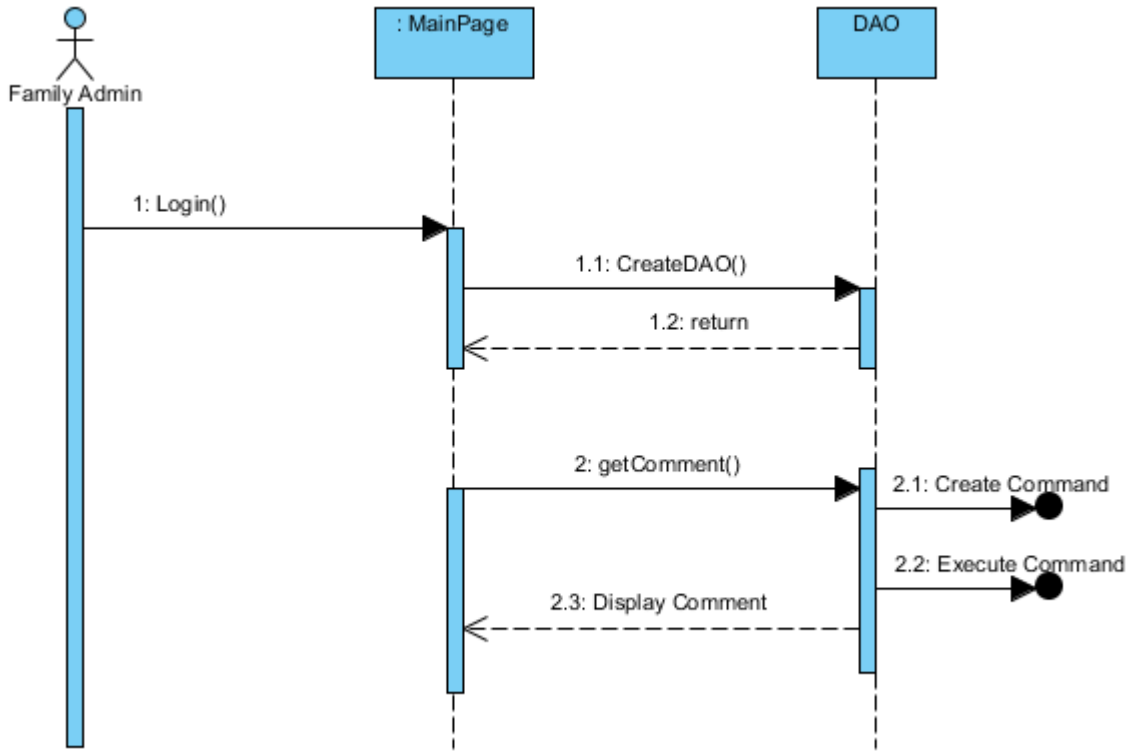
4.5.9. For use case "*Delete Tree*"



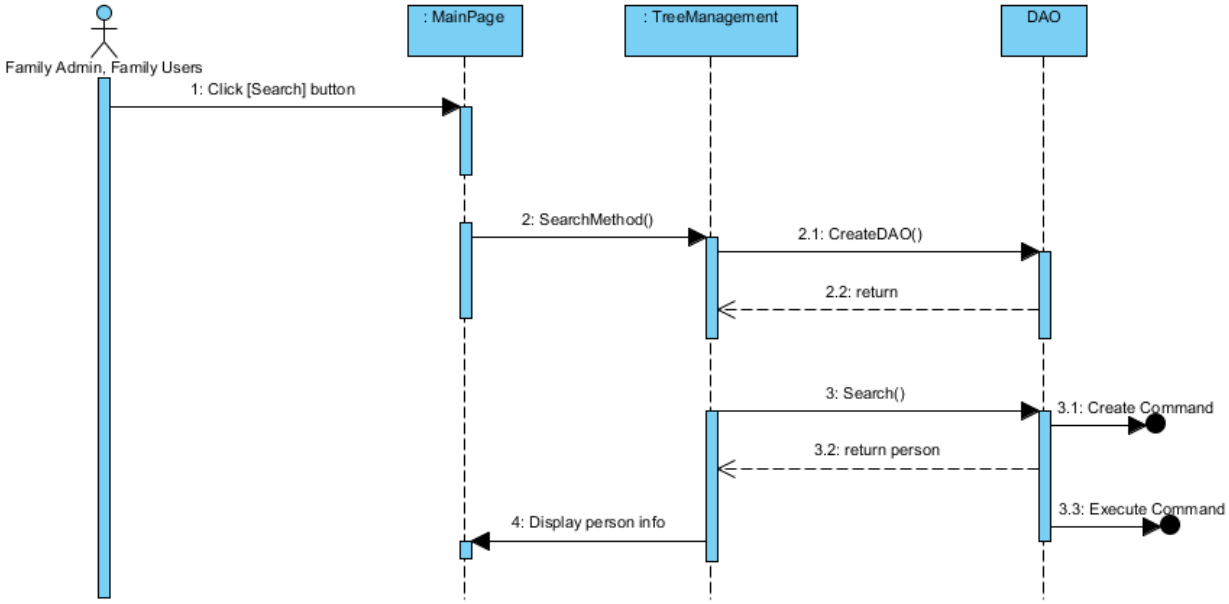
4.5.10. For use case "*Send Comment*"



4.5.11. For use case "*Review Comment*"



4.5.12. For use case "*Search*"



4.6. User Interface Design/Hardware Interface Design

4.6.1. Description of the User Interface

This section will describe the functioning of the program by discussing the main components of the screen.

- System Administrator

  The Menu bar contains the following options:

  1. Home Page
  2. About
  3. Manage User (list of user in Family Tree system)
  4. Manage Tree (list of family tree)
  5. Notification (receive request from family administrators)
- Family Administrator

  The Menu bar contains the following options:

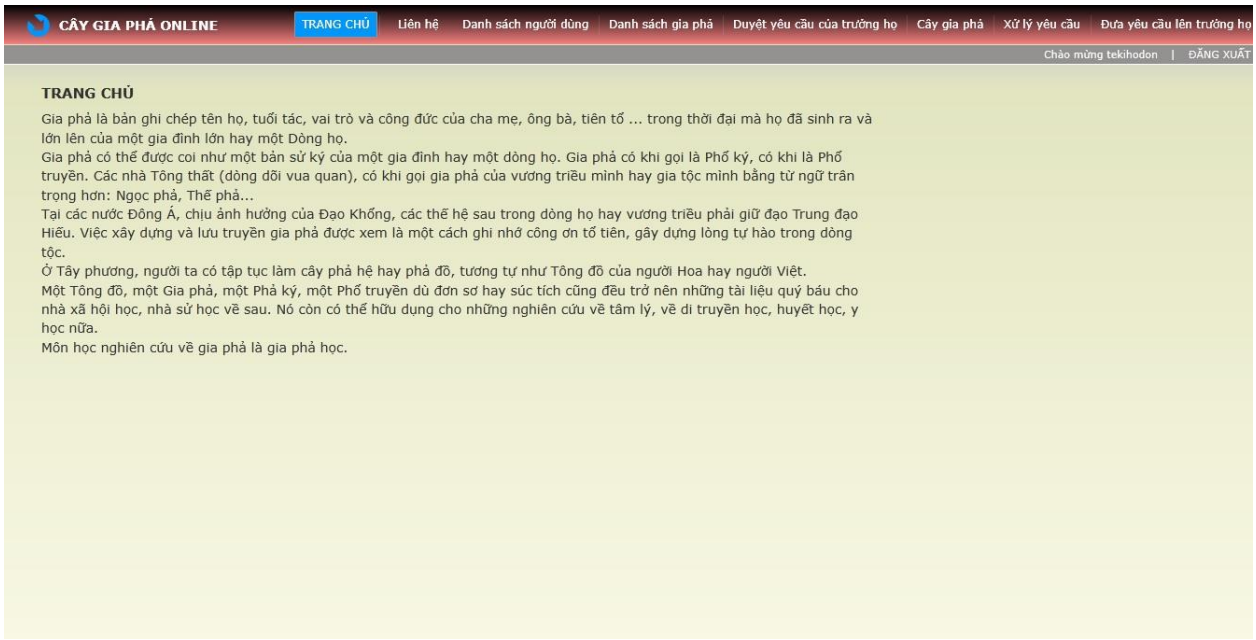
  1. Home Page
  2. About
  3. Family Tree
  4. Notification (receive request from family users, approve family users)
- Family user

  The Menu bar contains the following options:

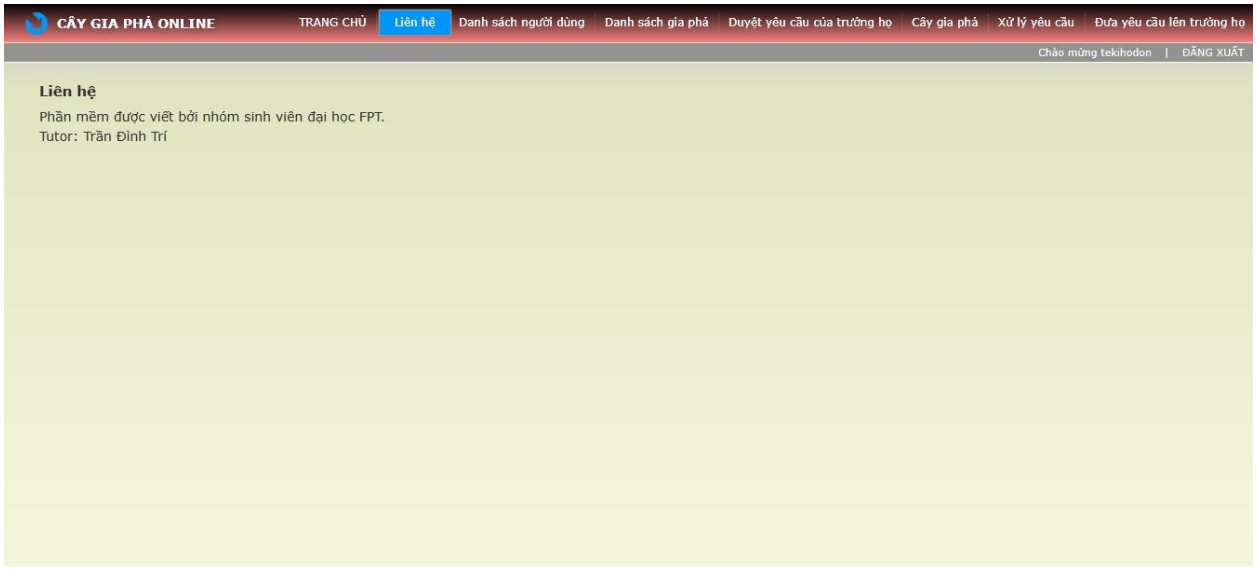
  1. Home Page
  2. About
  3. Family Tree
  4. Notification (send comment to family administrator)

4. 6.1.1. Screen Image/Hardware Description

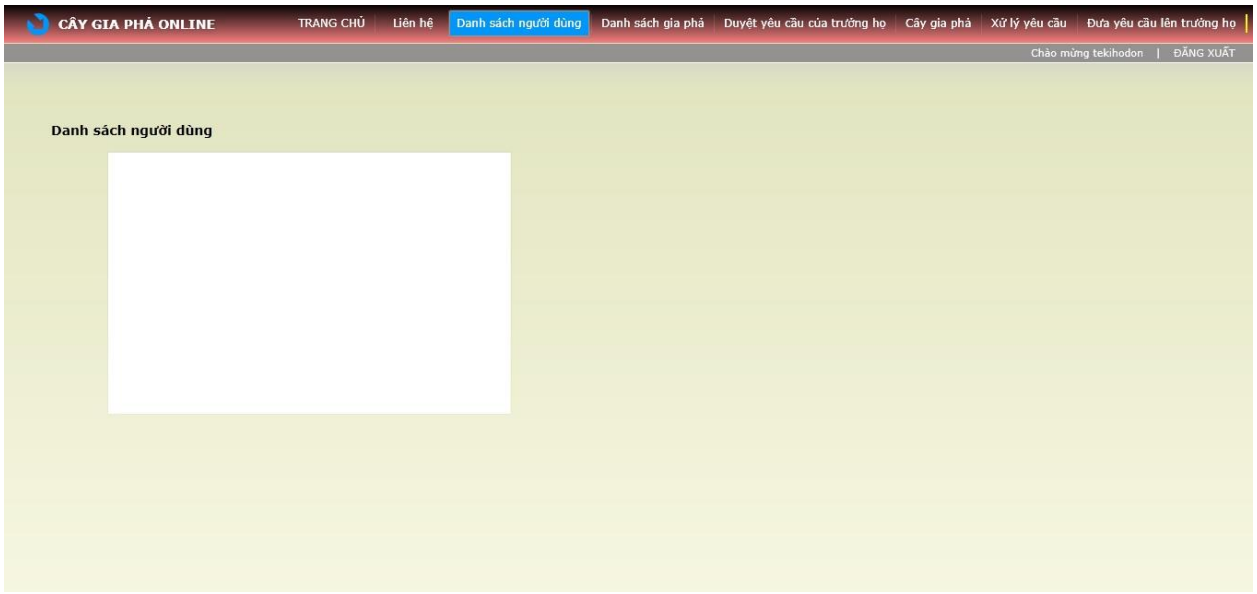
4.6.1.1.1. “Home Page” screen



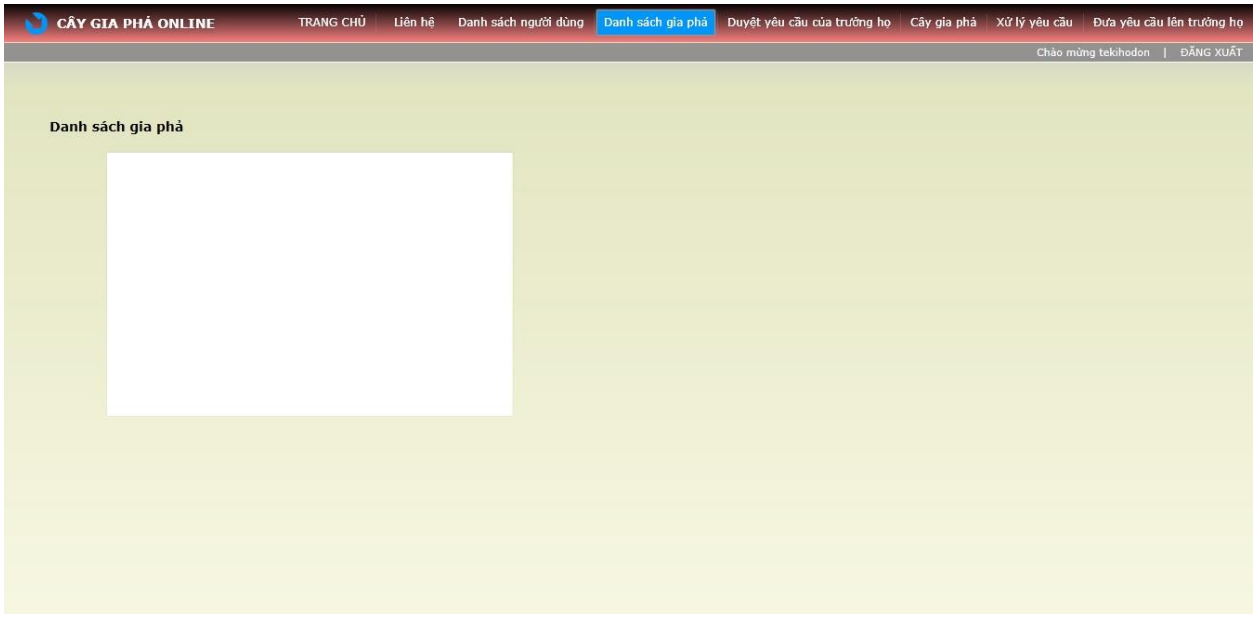
4.6.1.1.2. “About” screen



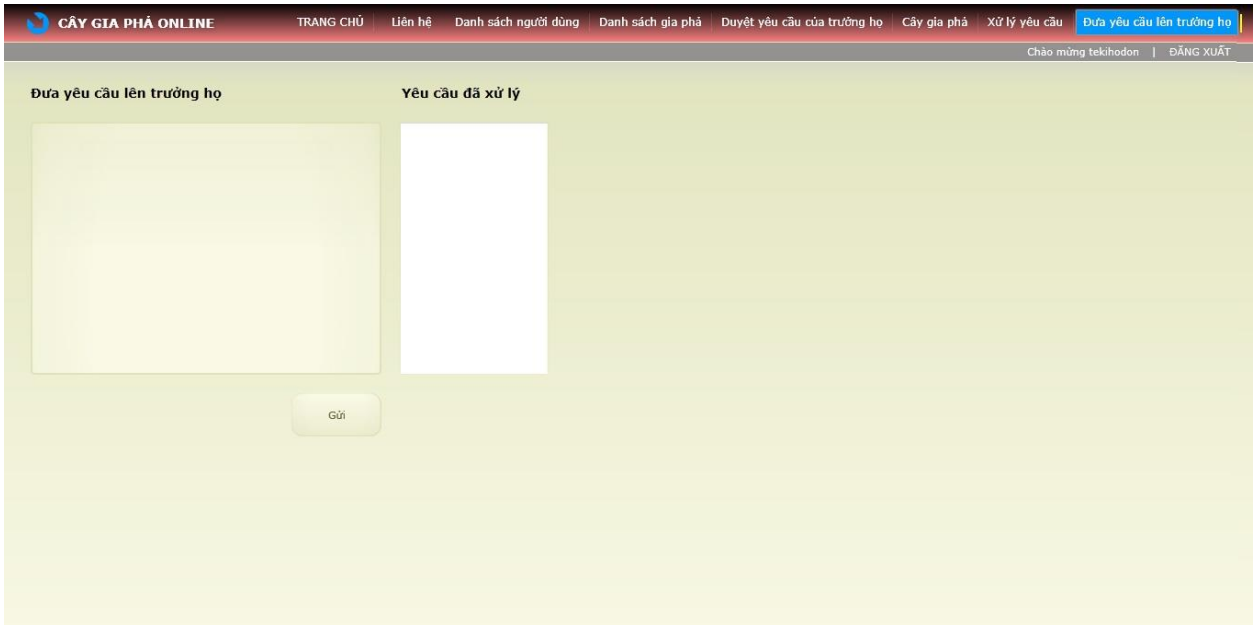
4.6.1.1.3. “Manage User” screen

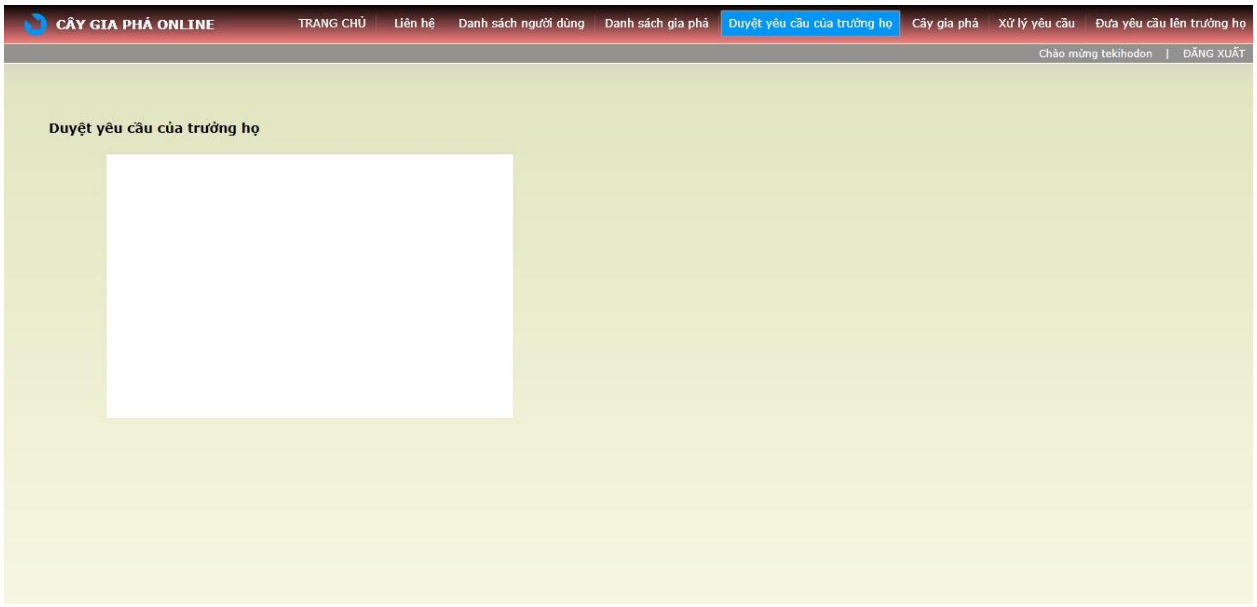


4.6.1.1.4. “Manage Tree” screen



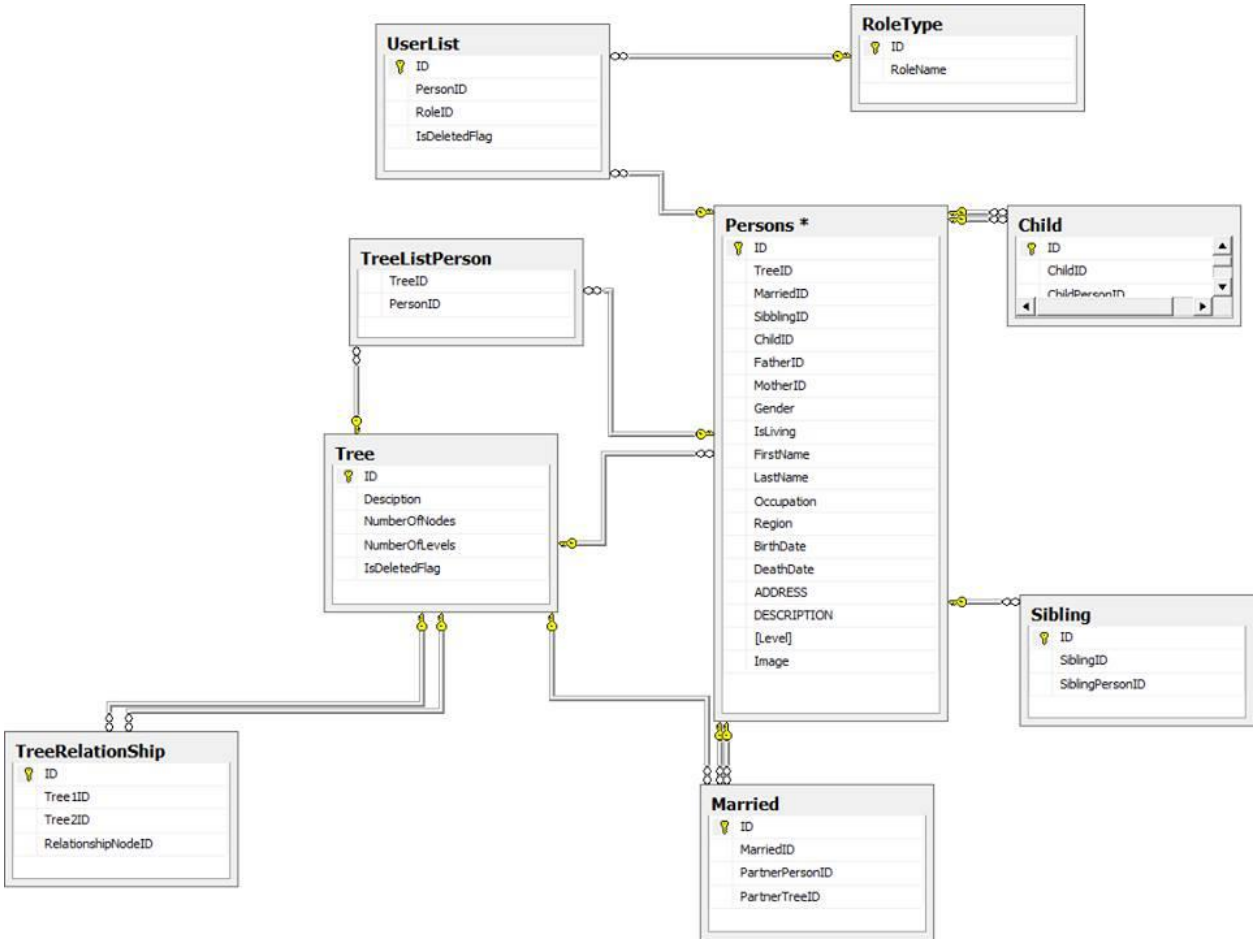
4.6.1.1.5. Notifications





4.7. Database Design or Data Structure or algorithms

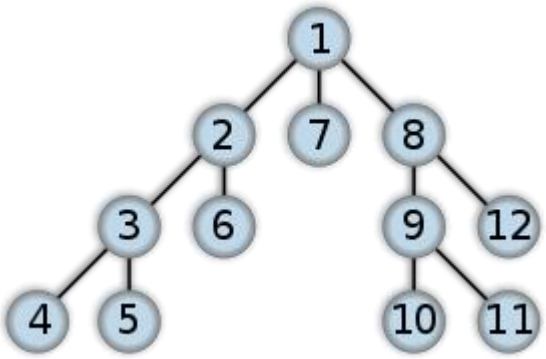
4.7.1. Database Design



4.7.2. Algorithms

4.7.2.1. Algorithm 1: DFS (Depth-first search)

Depth-first search (**DFS**) is an algorithm for traversing or searching tree or graph data structures. One starts at the root (selecting some node as the root in the graph case) and explores as far as possible along each branch before backtracking.



4.7.2.2. Algorithm 2: Recursive tree construction algorithms

4.7.2.2.1. CraftTree function

```
Variables: Node FirstNode, Tree master

     CraftTree(Node atNode)
           if (this node's FatherID is null
              AND this node's MotherID is null)
               At the highest node:
                   Check for Partners
           Else
                Recursively Traverse to highest node
                CraftTree(This node 's Father node)
```

4.7.2.2.2. Load Partner function

```
      Load_Partner(Node atNode)
           partners = atNode's List of partners
           Clear all partners in atNode
           foreach person in partners

              var partnerNode = NodeCache[person.ID]
               if (partnerNode.person has Children.Count > 0)
                   Load_Children_Partner(
                      atNode's person ID, partnerNode)

               master.AddPartner(atNode, partnerNode )

           firstNode = atNode
```

4.7.2.2.3. Load Children-Partner function

```
Load_Children_Partner(string currentID, Node partnerNode)

    foreach pair of partner_children in
                    partnerNode.person.Children

        if partner_children.Key equals currentID
            var children = partner_children.Value

            foreach childID in children collection
                var child = get node from NodeCache
                                        has key = childID

                if child.person.Partners.Count > 0
                   Load_Partner(child)

                var father = get node from NodeCache that
                                        has child.person.FatherID

                var mother = get node from NodeCache that
                                        has child.person.MotherID

                this.master.AddChildren(father, mother, child)
```

Chapter 5: Software Test Document

5.1. Introduction

5.1.1. System Overview

The primary focus of this plan is to ensure that the website can collect information from sources, database, display correctly information about family tree and provide advanced functions wrote in requirement.

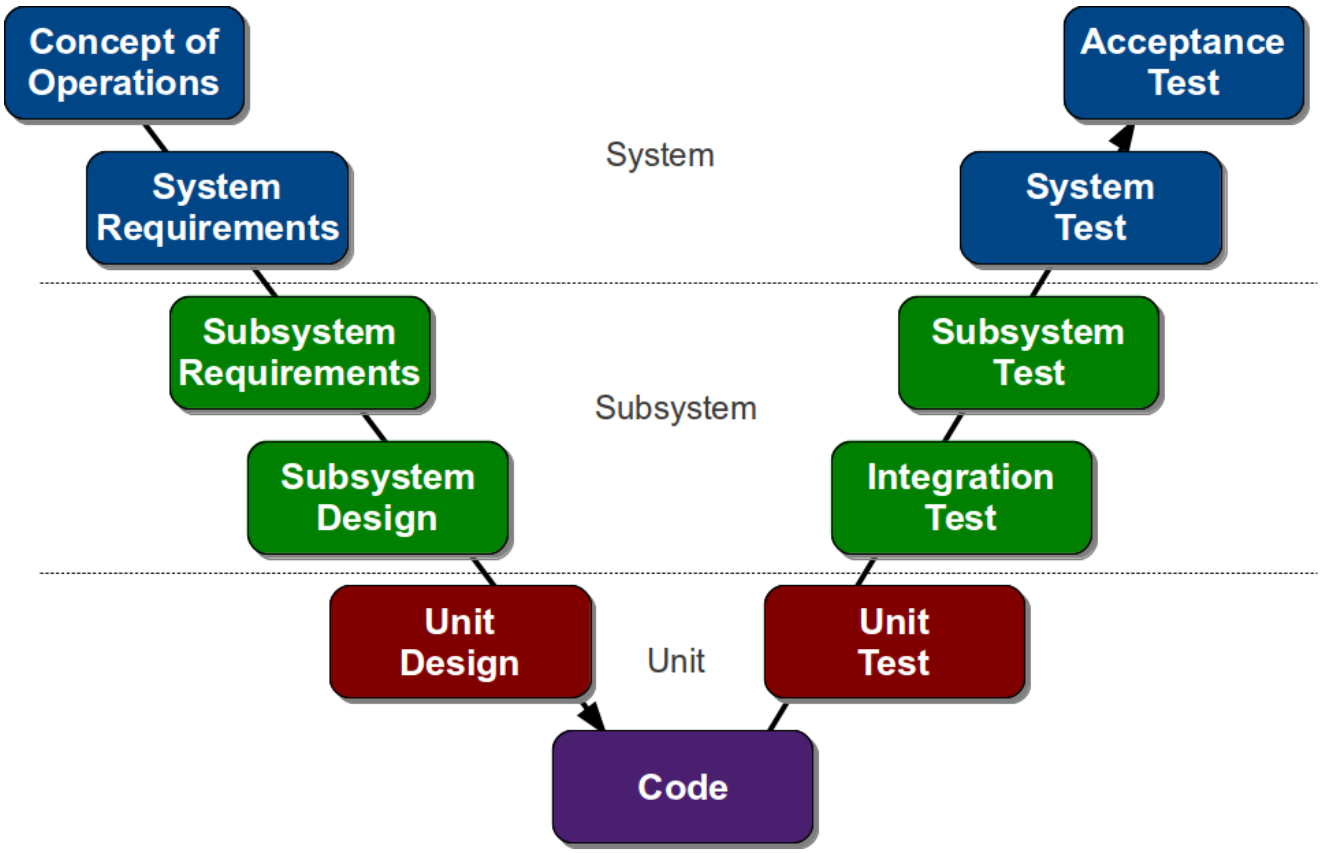
Guarantee the system can be run normally, and all requirements are correctly.

It means this website have to:

- Passed the stages of testing: Unit, Integration/System, and User Acceptance test
- Passed the types of testing: Interface, Function and Performance is good (do not so slow)
- Run normally in required browsers.

5.1.2. Test Approach

"Family Trees" project follows V-Model process, so we define or phases of testing associated with phases of development life-cycle.



The project will have three levels of testing, Unit, Integration/Subsystem and Acceptance/System Test. The details for each level are addressed in the approach section and will be further defined in the level specific plans.

"Family Trees" will be tested by 4 phases:

Phase 1: Unit test

- Unit testing will be done by developers and be approved by team leader.
- When executing unit test case, if any bugs are found, developers have to log and fix it until it correct.
- The unit test case file will be reviewed and passed by tester of the project.
- Unit testing is also the information provided to tester to perform Integration test and System Test
- Rule for filling test result:

Test result pass	Pass
Test result fail	Fail
Do not test	Untested
Cannot test	N/A (Not available)

Phase 2: Integration test

- After finish unit testing, integration test will be performed by tester of the project.
- Material is test cases and test tools.
- Do test by flow of functions and items which have concern each other.
- When executing Integration test case, if any bugs are found, tester have to log to "DefectTrackingLog" file and assign to developer fix it and redo this process until it correct.
- Rule for filling test result:

Test result pass	Pass
Test result fail	Fail
Do not test	Untested
Cannot test	N/A (Not available)

Phase 3: System test

- After finish Integration testing and developers collect all functions and items, tester will be performed system test, it means doing test whole system.
- If any bugs are found, developers have to fix and tester will verify them. System test is ended only when test cases are passed and no bug is found.
- For reduce testing time, the system test will be execute as free test.

Phase 4: Acceptance Test

- Base on requirement specification, system is tested again, for ensure there is not lacking or mistake any requirement.
- If there is any problem, developers have to fix/update and tester will verify them.
- Acceptance test is ended only when whole system maps requirement specification.

5.2. Test plan

5.2.1. Feature to be tested

5.2.1.1.Items

5.2.1.1.1. System Administrator functions

- Login to website
- Create accounts
- View information: trees, user list, user information…
- Edit user account
- Approve family administrator's request
- Manage trees (edit/show/restore/delete/search/comment/delete)

5.2.1.1.2. Family Administrator functions

- Login to website
- Create accounts
- View information: trees, user list, user information…
- Edit user account
- Manage tree (edit/show/restore/delete/search/comment/delete/expand/print/customize)
- Approve family user's request
- Create tree
- Create nodes
- Manage nodes (edit/view/comment/delete)
- Notification: review comment from family user

5.2.1.2. Family User function

- Login to website
- Create accounts
- View a tree
- Search person
- Search relationship between two people in a tree
- Print tree
- Expand tree
- Notification: send comment to family administrator

5.2.2. Feature not to be tested

- The stable of website when do not connect internet.
- Too much users connect on system.
- Test on too much web browser tools or mobile devices.

5.2.3. Testing Tool and Environment

5.2.3.1. Testing tool

List tools will be employed for this project:

Purpose	Tool	Vendor/In-house	Version
Defect log	MS-Excel		
Test effort	MS-Excel		
Document management	Assembla		

5.2.3.2. Environment

a. Test client configuration:

+ Computer: Internet Explorer 8, Chrome 20+, Firefox 12+, Safari

b. Software

+ Operating system: Windows 7
+ Framework: .Net 4.5 Framework
+ IDE: Visual Studio For Web 2012
+ DBMS: SQL Server 2012 Express

5.2.3.3. Test strategy

5.2.3.4. Test types

+ Function Testing
+ User Interface Testing
+ Performance testing
+ Security and Access Control Testing

5.2.3.5. Test stages:

Clearly state the stage in which the test will be executed. Identified below are the stages in which common test are executed:

Type of Tests	Stage of Test			
	Unit	Integration	System	Acceptance
Function Test	X	X	X	X
User Interface test	X	x	x	x
Performance Tests	X		x	

Type of Tests	Stage of Test			
	Unit	Integration	System	Acceptance
Volume test	x		X	
Security test	X		X	

5.2.3.6. Risk list

- Lack of experience about web-service -> Study
- Deployment, test environment -> Provide
- Performance test: Cannot test the case which so many user connect to website
- Change request: Proposed changes will be discussed with all members of the project. The project leader will determine the impact of the change. After that, it will be implemented.
- Server for test

5.2.3.7. Testing document management

5.2.3.8. Resource and responsibilities

This table shows the staffing assumptions for the project.

Worker/Doer	Specific Responsibilities/Comments
(Test Leader) DungNP	- Manage Test resource and assign test tasks - Create TC for modules : System Administrator, Family Administrator, Family User - Execute test: System Administrator - Create test report: SST of all modules
DungNP	- Create TC for modules : All modules - Execute test: System Administrator - Report test result: All report
HuyenTTT	- Create TC for modules : All modules - Execute test: Family administrator, Family User - Report test result: All report

5.2.3.9. Test milestones

The estimated time line for this project is very aggressive (4 months).

Milestone Task	Effort (pd)	Start Date	End Date
Create Unit Test case	14	2013/07/09	2013/07/23
Review & update UTC	14	2013/07/09	2013/07/23
Execute Unit test case	14	2013/07/09	2013/07/23
Create Test Plan	3	2013/07/09	2013/07/12
Review & update TP	3	2013/07/12	2013/07/13

Create Integration Test case	14	2013/07/13	2013/07/27
Review & Update Integration TC	8	2013/07/14	2013/07/22
Execute Integration test	14	2013/07/23	2013/08/06
Execute System test	*6*	2013/07/25	2013/07/31
Execute Acceptance test	8	2013/07/31	2013/08/08

5.2.3.10. Deliverables

No	Deliverables	Delivered Date	Delivered by	Delivered to
1	Test Plan	2013/07/12	HuyenTTT	Whole team
2	Unit Test cases	2013/07/23	HuyenTTT DungNP	Whole team
3	Integration Test Cases	2013/07/27	DungNP HuyenTTT	Whole team
4	Defect log	Update daily	HuyenTTT	Whole team
5	Test reports	After finish testing	DungNP	Whole team

5.3. Test cases

5.3.1. Purpose

Confirm that website can run normally and finding all of bugs to developer can fix them.

5.3.2. Inputs

- User requirement
- Software requirement
- Software architect design

5.3.3. Expected Outputs & Pass/Fail criteria

Criteria for Unit test of Development team, for Test team accepts to start testing:

- Number of UTC/KLOC: 60 ~ 80 UTC/KLOC
- Number defects/KLOC: 5~6 defects/KLOC
- Statement coverage: 100%
- Branch coverage: 100%
- Path coverage: 100%

Criteria which are based on to accept the products:

- All the inputs of system testing are baselined and approved by relevant stakeholders (Common requirement, ADD, SRS, Design,Software package, System test cases).
- Test coverage: 100%
- Successful Test coverage: 95%
- Number of Test cases (Unit/System Test cases): 129 TCs/KLOC
- Number of defects/Weighted defects: 3~4 defects/KLOC
- System testing is over and all known issues (if any) are aware by relevant stakeholders (Team lead, Test lead, PM, Tech lead) and documented in Test summary report.
- Only Cosmetic defects can be accepted

5.3.4. Test Procedure

- White box test: through execute Unit test case
- Black box test: through execute System test case

5.4. Checklists

5.4.1. Checklist of Validation

For detail of each checklist: Refer to checklist of Unit test case

No	Checklist	Create date	Create by	Status
1	FT2013_Checklist_SST_UnitTestcase_FamilyAdministrator	7/23/2013	TrungNT	OK
2	FT2013_Checklist_SST_UnitTestcase_SystemAdministrator	7/23/2013	LongTD	OK
3	FT2013_Checklist_SST_UnitTestcase_FamilyUser	7/23/2013	DucPD	OK
4	FT2013_Checklist_SST_UnitTestcase_Login	7/23/2013	DucPD	OK

5.4.2. Submission Checklist

For detail of each checklist: Refer to checklist of System test case

No	Checklist	Create date	Create by	Status
1	FT2013_Checklist_SST_Testcase_FamilyAdministrator	7/15/2013	HuyenTTT	OK
2	FT2013_Checklist_SST_Testcase_SystemAdministrator	7/15/2013	DungNP	OK
3	FT2013_Checklist_SST_Testcase_FamilyUser	7/15/2013	HuyenTTT	OK
4	FT2013_Checklist_SST_Testcase_Login	7/15/2013	DungNP	OK

5.4.3. Other material

5.4.3.1. Log for test

About content of defect, refer to file: “FT2013_DefectList_V1.0” by Defect number

5.4.3.2. Test Results

5.4.3.2.1 Unit test case

Component	Class	Method	Status	Date	Tester
Connector	Connector	CheckExistence()	OK	7/23/2013	TrungNT
		ChildrenConnection()	OK	7/23/2013	TrungNT
		Connector()	OK	7/23/2013	TrungNT
		Connector	OK	7/23/2013	TrungNT
		Dispose	OK	7/23/2013	TrungNT
		Hide	OK	7/23/2013	TrungNT
		PartnerConnection	OK	7/23/2013	TrungNT
		Show	OK	7/23/2013	TrungNT
		UpdateLocation	OK	7/23/2013	TrungNT
CraftEngine	CraftEngine	CraftEngine	Neutral	7/23/2013	TrungNT
		CraftTree	Neutral	7/23/2013	TrungNT
		Load_Children_Partner	OK	7/23/2013	TrungNT
		Load_Husband	OK	7/23/2013	TrungNT
		Load_Partner	OK	7/23/2013	TrungNT
		LoadTreeOf	OK	7/23/2013	TrungNT
		NodeCacheGenerator	OK	7/23/2013	TrungNT
DFSEngine	DFSEngine	DFSEngine	OK	7/23/2013	TrungNT
		FetchData	OK	7/23/2013	TrungNT
		Search	OK	7/23/2013	TrungNT
DynamicListBox	DynamicListBox	createButton	OK	7/23/2013	TrungNT
		DynamicListBox	OK	7/23/2013	TrungNT
		LoadCouple	OK	7/23/2013	TrungNT
		SelectedItem_Click	OK	7/23/2013	TrungNT
GraphicLibrary.Connection	GraphicLibrary.Connection	ChildrenConnection	OK	7/23/2013	TrungNT
		PartnerConnection	OK	7/23/2013	TrungNT
Master	Master	ActivateTree	OK	7/23/2013	TrungNT
		AddChildren	OK	7/23/2013	TrungNT
		AddConnector(Connector)	OK	7/23/2013	TrungNT
		AddDaughter(Node, Node, String)	OK	7/23/2013	TrungNT
		AddFather(Node, string)	OK	7/23/2013	TrungNT
		AddFirstNode(Node)	OK	7/23/2013	TrungNT
		AddMother (Node, string)	OK	7/23/2013	TrungNT
		AddParent	Neutral	7/23/2013	TrungNT
		AddPartner	Neutral	7/23/2013	TrungNT
		AddRelation	OK	7/23/2013	TrungNT
		AddWife	OK	7/23/2013	TrungNT
		Caching	OK	7/23/2013	TrungNT
		CollapseAllTree	OK	7/23/2013	TrungNT
		CraftChildren	OK	7/23/2013	TrungNT

		CraftParent	OK	7/23/2013	TrungNT
		CraftPartner	OK	7/23/2013	TrungNT
		CreateTree	OK	7/23/2013	TrungNT
		CurrentActiveTree	OK	7/23/2013	TrungNT
		DrawConnection	OK	7/23/2013	TrungNT
		ExportAllPerson	OK	7/23/2013	TrungNT
		FetchData	OK	7/23/2013	TrungNT
		GrowBottom	OK	7/23/2013	TrungNT
		GrowTop	OK	7/23/2013	TrungNT
		GUID	OK	7/23/2013	TrungNT
		Init	OK	7/23/2013	TrungNT
		Master	OK	7/23/2013	TrungNT
		RemoveNode	OK	7/23/2013	TrungNT
		RemoveNode	OK	7/23/2013	TrungNT
		RemoveTree	OK	7/23/2013	TrungNT
		SaveClear_CurrentTree	OK	7/23/2013	TrungNT
		Uncaching	OK	7/23/2013	TrungNT
		Uncaching	OK	7/23/2013	TrungNT
Person	Person	AddChild	OK	7/23/2013	TrungNT
		AddParent	OK	7/23/2013	TrungNT
		AddPartner	OK	7/23/2013	TrungNT
		Person	OK	7/23/2013	TrungNT
		Person	OK	7/23/2013	TrungNT
InfoPanel	InfoPanel	AddImageButton_Click	OK	7/23/2013	TrungNT
		CancelButton_Click	OK	7/23/2013	TrungNT
		ChoosePicture	OK	7/23/2013	TrungNT
		DisplayInfo	OK	7/23/2013	TrungNT
		Hide	OK	7/23/2013	TrungNT
		InfoPanel	OK	7/23/2013	TrungNT
		InitializeComponent	OK	7/23/2013	TrungNT
		SaveButton_Click	OK	7/23/2013	TrungNT
		SaveInfo	OK	7/23/2013	TrungNT
		Show	OK	7/23/2013	TrungNT
MessagePanel	MessagePanel	Hide	OK	7/23/2013	TrungNT
		MessagePanel	OK	7/23/2013	TrungNT
		InitializeComponent	OK	7/23/2013	TrungNT
Node	Node	AddChild	OK	7/23/2013	TrungNT
		AddPartner	OK	7/23/2013	TrungNT
		Alignment	OK	7/23/2013	TrungNT
		ChangeGender	OK	7/23/2013	TrungNT
		DecreaseFixedPoint	OK	7/23/2013	TrungNT
		DecreaseSpace	OK	7/23/2013	TrungNT
		Edit	OK	7/23/2013	TrungNT
		Highlight	OK	7/23/2013	TrungNT

		IncreaseFixedPoint	OK	7/23/2013	TrungNT
		IncreaseSpace	OK	7/23/2013	TrungNT
		Init	OK	7/23/2013	TrungNT
		InitializeComponent	OK	7/23/2013	TrungNT
		MakeSpace	OK	7/23/2013	TrungNT
		Node	OK	7/23/2013	TrungNT
		Node	OK	7/23/2013	TrungNT
		Node	OK	7/23/2013	TrungNT
		NodeBorder_MouseEnter	OK	7/23/2013	TrungNT
		NodeBorder_MouseLeave	OK	7/23/2013	TrungNT
		RegisterHandler	OK	7/23/2013	TrungNT
		Release	OK	7/23/2013	TrungNT
PopupPanel	PopupPanel	ApplyEmbeddedSetting	OK	7/23/2013	TrungNT
		ClearChoice	OK	7/23/2013	TrungNT
		GetItemList	OK	7/23/2013	TrungNT
		Hide	OK	7/23/2013	TrungNT
		InitializeComponent	OK	7/23/2013	TrungNT
		LoadFor	OK	7/23/2013	TrungNT
		PopupPanel	OK	7/23/2013	TrungNT
		PostStatus	OK	7/23/2013	TrungNT
		ReleaseFromParent	OK	7/23/2013	TrungNT
		Show	OK	7/23/2013	TrungNT
		Show	OK	7/23/2013	TrungNT
RenderScreen	RenderScreen	ActivateTree	OK	7/23/2013	TrungNT
		AddButton_Click	OK	7/23/2013	TrungNT
		Canvas_MouseLeave	OK	7/23/2013	TrungNT
		CreateDynamicBox	OK	7/23/2013	TrungNT
		CreateNewNode	OK	7/23/2013	TrungNT
		CreateTreeButton_Click	OK	7/23/2013	TrungNT
		DebugTree	OK	7/23/2013	TrungNT
		DrawButton_Click	OK	7/23/2013	TrungNT
		EditButton_Click	OK	7/23/2013	TrungNT
		EventBinding	OK	7/23/2013	TrungNT
		InitializeComponent	OK	7/23/2013	TrungNT
		MessageButton_Click	OK	7/23/2013	TrungNT
		Node_Click	OK	7/23/2013	TrungNT
		NodeTypeBox_SelectionChanged	OK	7/23/2013	TrungNT
		Pan	OK	7/23/2013	TrungNT
		ProcessPersonInfo	OK	7/23/2013	TrungNT
		ProcessSearchResult	OK	7/23/2013	TrungNT
		RemoveButton_Click	OK	7/23/2013	TrungNT
		RenderScreen	OK	7/23/2013	TrungNT
		ReverseHighlight	OK	7/23/2013	TrungNT

		Screen_KeyDown	OK	7/23/2013	TrungNT
		Screen_MouseMove	OK	7/23/2013	TrungNT
		Screen_MouseRightButton Down	OK	7/23/2013	TrungNT
		Screen_MouseRightButton Up	OK	7/23/2013	TrungNT
		Screen_MouseWheel	OK	7/23/2013	TrungNT
		SearchButton_Click	OK	7/23/2013	TrungNT
		TaskBar_MouseEnter	OK	7/23/2013	TrungNT
		TaskBar_MouseLeave	OK	7/23/2013	TrungNT
		TreeChanged	OK	7/23/2013	TrungNT
		UserControl_MouseMove	OK	7/23/2013	TrungNT
		ZIndexConfig	OK	7/23/2013	TrungNT
SearchPanel	SearchPanel	Hide	OK	7/23/2013	TrungNT
		InitializeComponent	OK	7/23/2013	TrungNT
		LoadCache	OK	7/23/2013	TrungNT
		NameFormat	OK	7/23/2013	TrungNT
		SearchButton_Click	OK	7/23/2013	TrungNT
		SearchPanel	OK	7/23/2013	TrungNT
Tree	Tree	Activate	OK	7/23/2013	TrungNT
		ActivateButton_Click	OK	7/23/2013	TrungNT
		AddNode	OK	7/23/2013	TrungNT
		ConfigTree	OK	7/23/2013	TrungNT
		Deactivate	OK	7/23/2013	TrungNT
		GeneralStatus	OK	7/23/2013	TrungNT
		InitializeComponent	OK	7/23/2013	TrungNT
		Length	OK	7/23/2013	TrungNT
		PostStatus	OK	7/23/2013	TrungNT
		RemoveNode	OK	7/23/2013	TrungNT
		RemoveNodeAt	OK	7/23/2013	TrungNT
		Space	OK	7/23/2013	TrungNT
		Tree	OK	7/23/2013	TrungNT
		UpdateID	OK	7/23/2013	TrungNT

5.4.3.2.2. Integration and System test case

Function	Screen	Viewpoint	Total	Normal	Abnormal	Boundary	Pass	Fail	N/A	Not tested	Create date	Create by	Execute date	Execute by	Status	Defect No
System Administrator	Browse	GUI										DungNP		DungNP		
														DungNP		
														DungNP		
		Perform										DungNP		DungNP		
														DungNP		
														DungNP		
	Index	GUI										DungNP		DungNP		
														DungNP		
		Perform										DungNP		DungNP		
														DungNP		
														DungNP		
	Search	GUI										DungNP		DungNP		
														DungNP		
														DungNP		
		Perform										DungNP		DungNP		

													DungNP		
													DungNP		
	View	GUI									DungNP		DungNP		
													DungNP		
													DungNP		
													DungNP		
		Perform									DungNP		DungNP		
													DungNP		
													DungNP		
FamilyAdministrator	Browse	GUI									HuyenTTT		HuyenTTT		
													HuyenTTT		
		Perform									HuyenTTT		HuyenTTT		
													HuyenTTT		
	Search	GUI									HuyenTTT		HuyenTTT		
													HuyenTTT		

		Perform										HuyenTTT		HuyenTTT		
														HuyenTTT		
	View	GUI										HuyenTTT		HuyenTTT		
														HuyenTTT		
		Perform										HuyenTTT		HuyenTTT		
Login	Login	GUI Perform										DungNP		DungNP		
														DungNP		
User	General	GUI										HuyenTTT		HuyenTTT		
														HuyenTTT		
		Perform										HuyenTTT		HuyenTTT		
														HuyenTTT		
	Dash board	GUI										HuyenTTT		HuyenTTT		
														HuyenTTT		
		Perform										HuyenTTT		HuyenTTT		
	Setting	GUI										HuyenTTT		HuyenTTT		
														HuyenTTT		
		Perform										HuyenTTT		HuyenTTT		
														HuyenTTT		

	Delete	GUI										HuyenTTT		HuyenTTT		
														HuyenTTT		
														HuyenTTT		
		Perform										HuyenTTT		HuyenTTT		
														HuyenTTT		
														HuyenTTT		
	Approve	GUI										HuyenTTT		HuyenTTT		
														HuyenTTT		
		Perform										HuyenTTT		HuyenTTT		
	Search	GUI										HuyenTTT		HuyenTTT		
		Perform										HuyenTTT		HuyenTTT		
	Report	GUI										HuyenTTT		HuyenTTT		
														HuyenTTT		
		Perform										HuyenTTT		HuyenTTT		
														HuyenTTT		
	Maintain	GUI										HuyenTTT		HuyenTTT		
		Perform										HuyenTTT		HuyenTTT		

5.4.3.2.3. Incident Report

Severity	Total Defect	Open Defect			
		Assigned	Corrected	Accepted	Total Open defects
Fatal					
Serious					
Medium					
Cosmetic					
Total (W.def)					

Chapter 6: Software User's Manual

6.1 Installation Guide

6.2 User's Guide

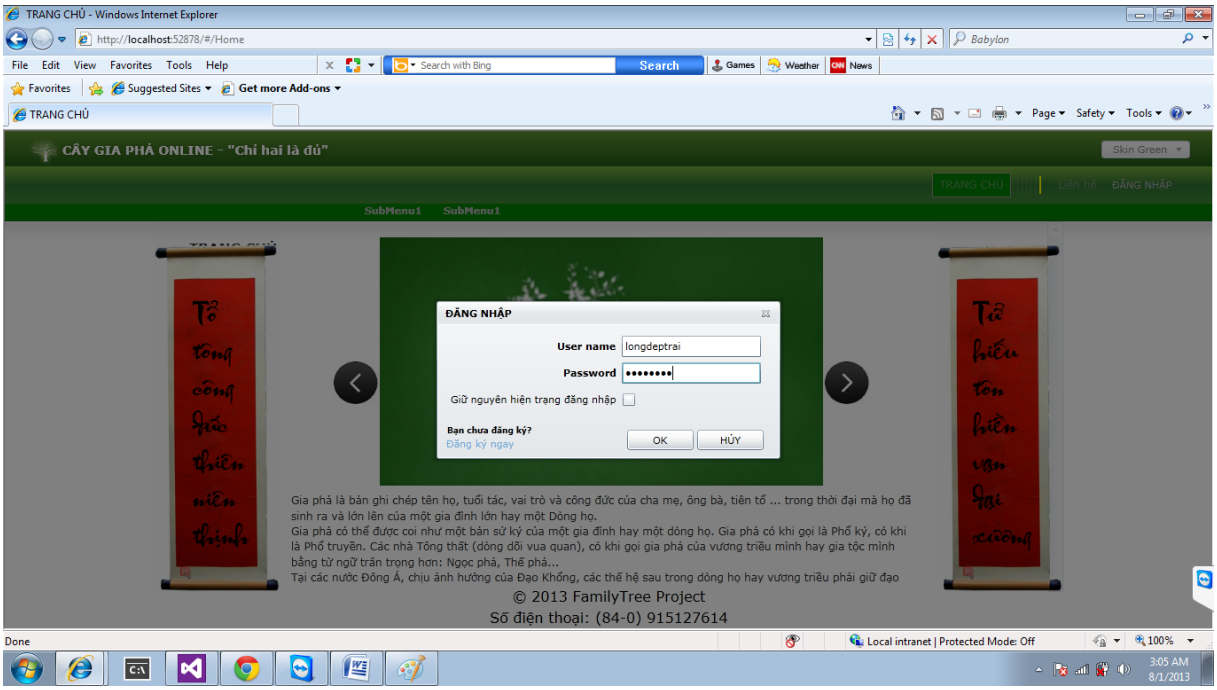
6.2.1. Family user Guide

6.2.1.1 Default

- When log into the website, user only see the "Home" page as below:

6.2.1.2 Login

- From the main page, user click the "Đăng nhập" hyperlink in the top left corner, log in screen will displayed as below:



- After loging into the system successfully, the screen will displayed that including the following items:

+ "Trang chủ": home page

+ "Cây gia phả": family tree after you have permission to access on this

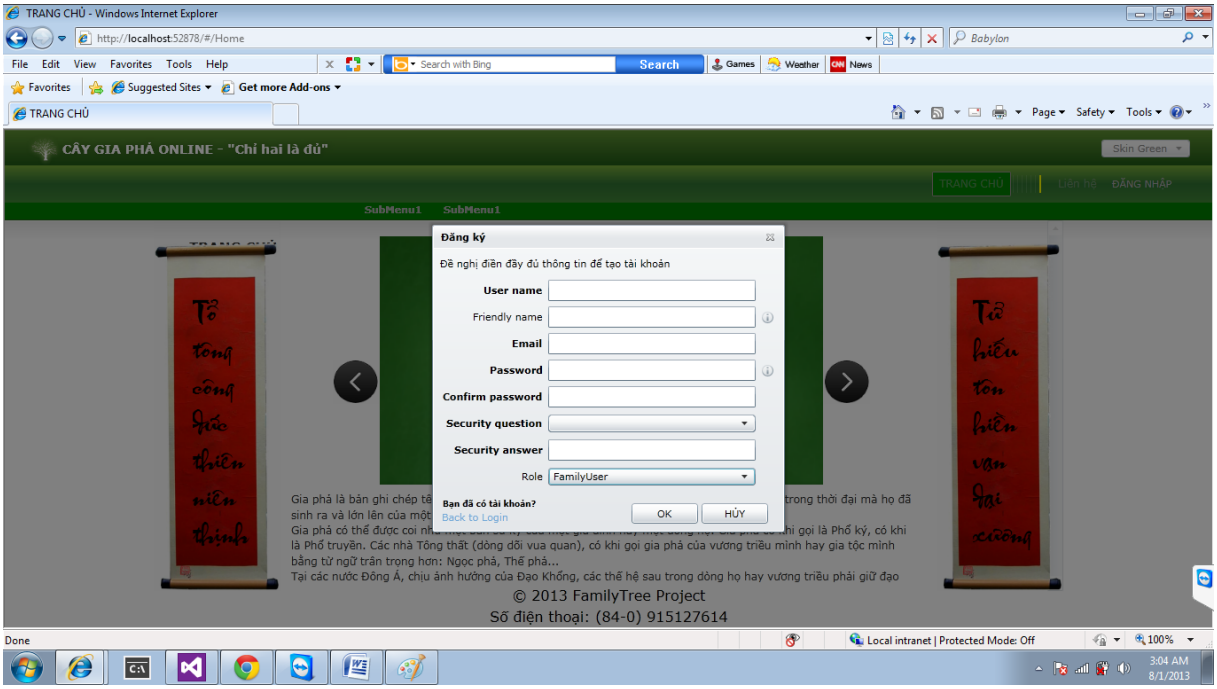
+ "Quản lý gia phả": List of family tree that has already created in the system

+ "Gửi yêu cầu": send comment to the family admin

+ "Liên hệ": displays some information about contact of development team

6.2.1.3 Create account

- If user has not account, click the "Đăng kí ngay" hyper link. This link is placed below the text "Bạn chưa đăng kí?" in the login screen, register account screen will displayed as:



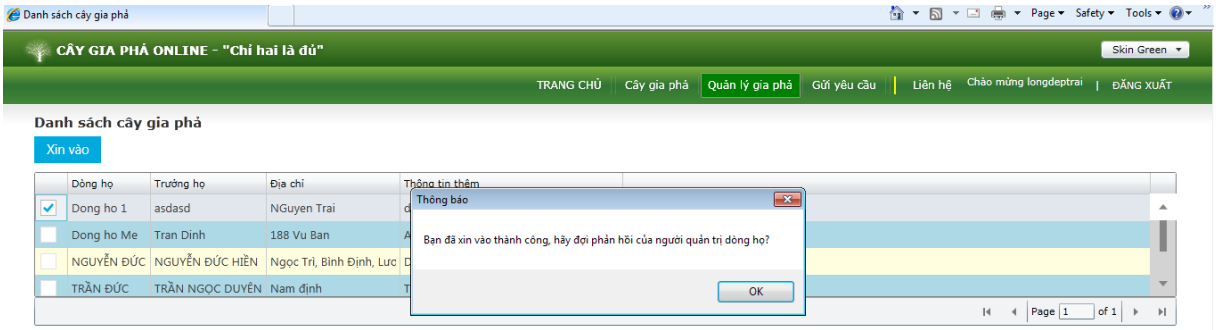
- Input the information into the fields in register screen, then click "OK" to create account, or click "HỦY" to cancel this action
- Please note that, you must select a role when creating an account.
  + If "Family admin" is selected, you can create, edit, delete, view tree(s), ….
  + If "Family user" is selected, you only can see the tree that has family admin's permission

6.2.1.4 View Tree

- After loging into the system as family user permission, you can do some action to view a tree
- Firstly, you can click the "Danh sách gia phả" tab in the menu bar to see all the family tree that has already created in the system

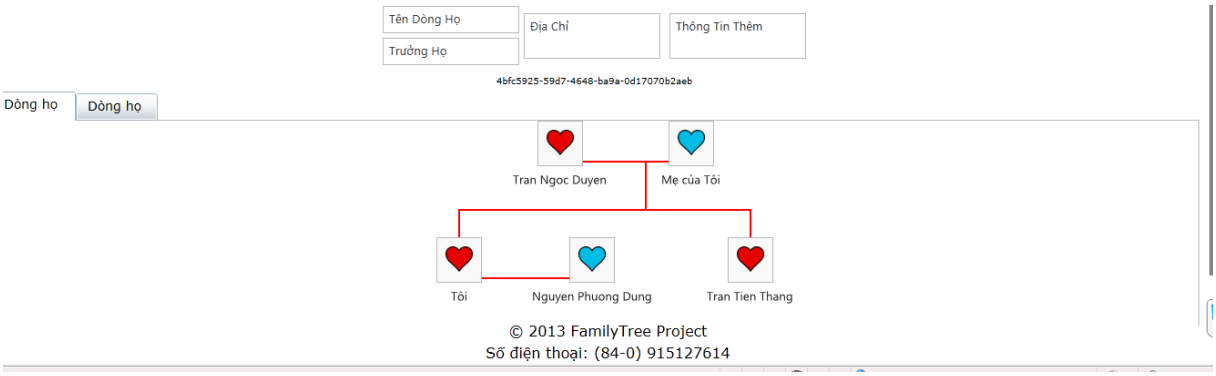
- Secondly, select a family tree that you want to see by mark in the check box

- Thirdly, click the “Xin vào” button, and then the information message will displayed:
- Information messge is “Bạn đã xin vào thành công, hãy đợi phản hồi của người quản trị dòng họ.”



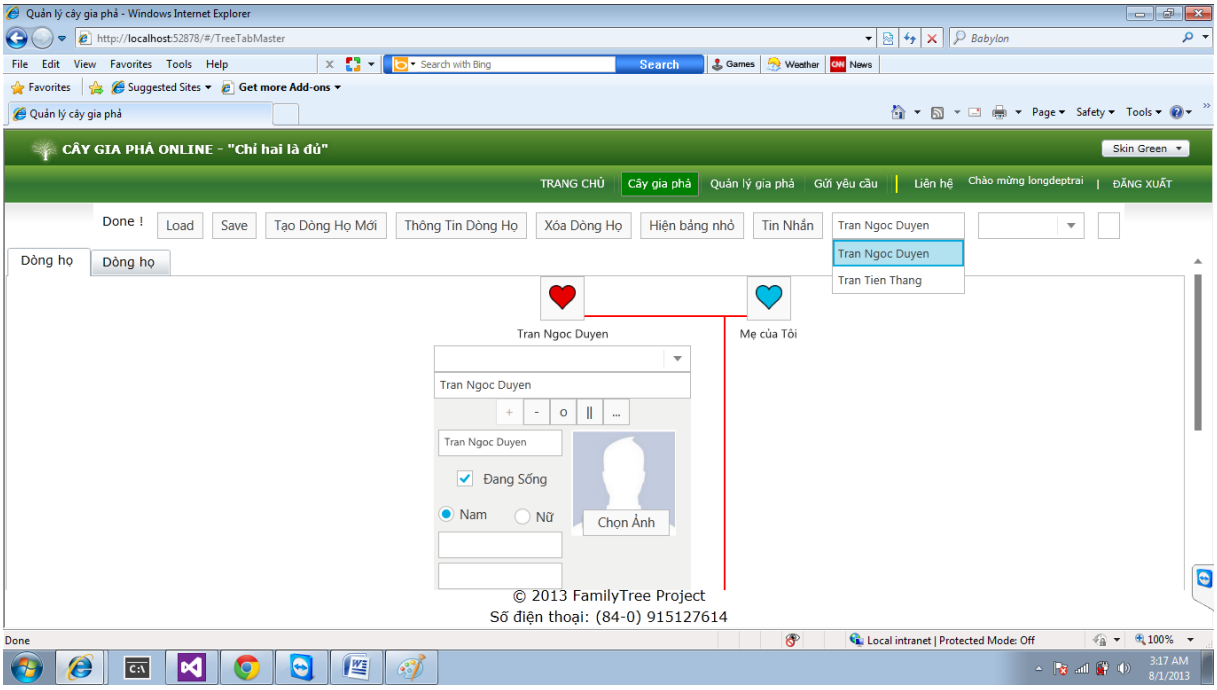
- Lastly, if family administrator has approved your request, click the “Cây gia phả” item in the menu bar, then you can see the family tree as your expected.

- In the “Cây gia phả” page, you can see family tree’s information by click the “Thông tin dòng họ” button



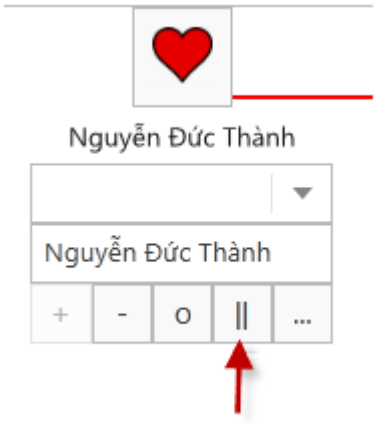
6.2.1.5 Search a person

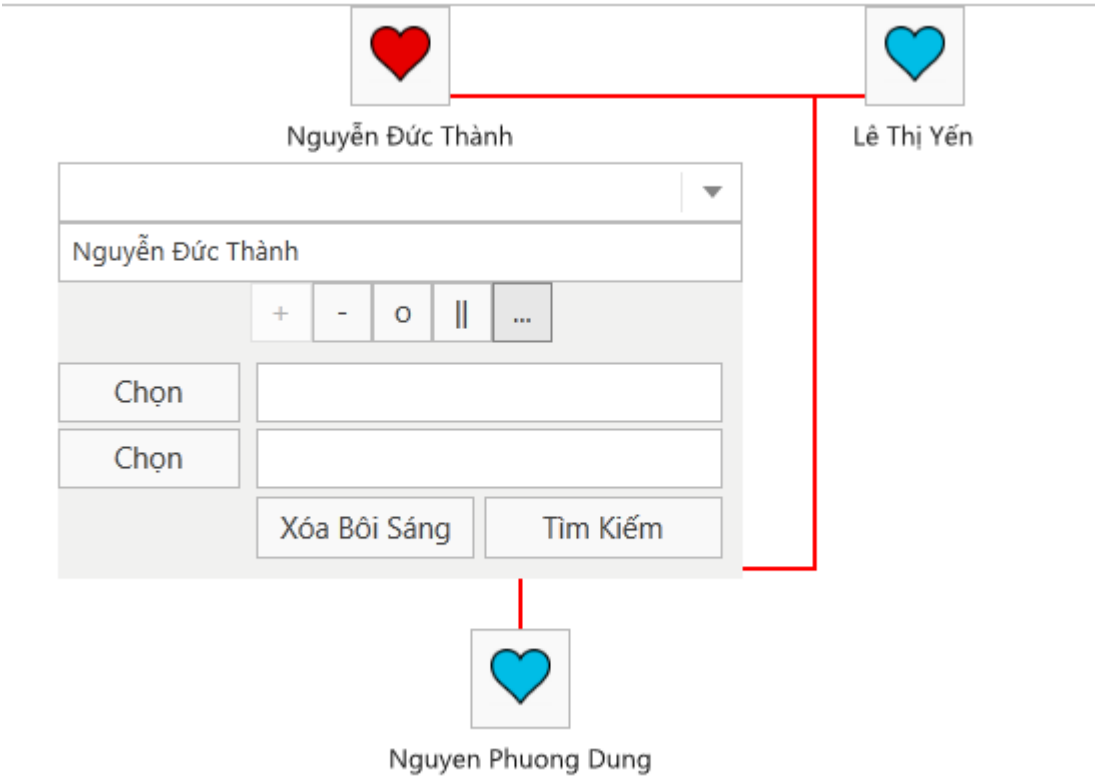
- In the "Cây gia phả" page, you can search a person in a tree by typing name that you want to search into the search text box. This text box is placed behind "Tin nhắn" button



6.2.1.6 Search relationship between 2 persons in a tree

- In the "Cây gia phả" tab, you can search relationsgip between 2 persons in a tree
  Step 1: Select a node, click the "||" icon to display search screen

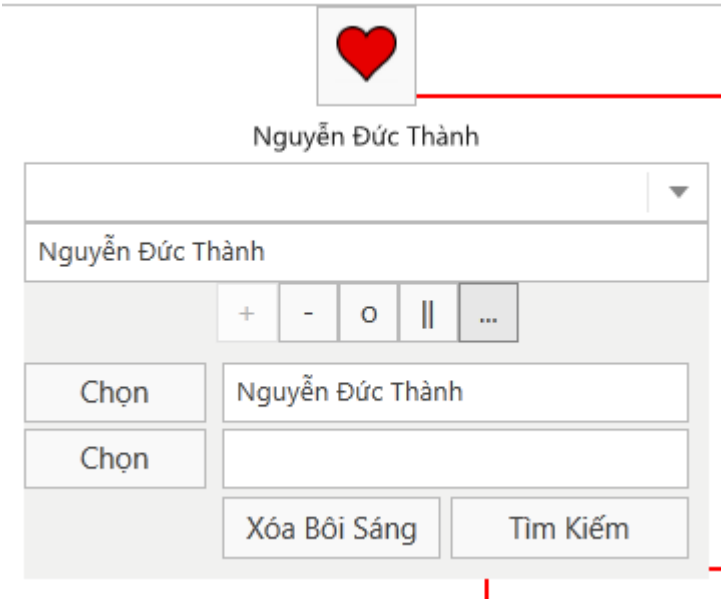




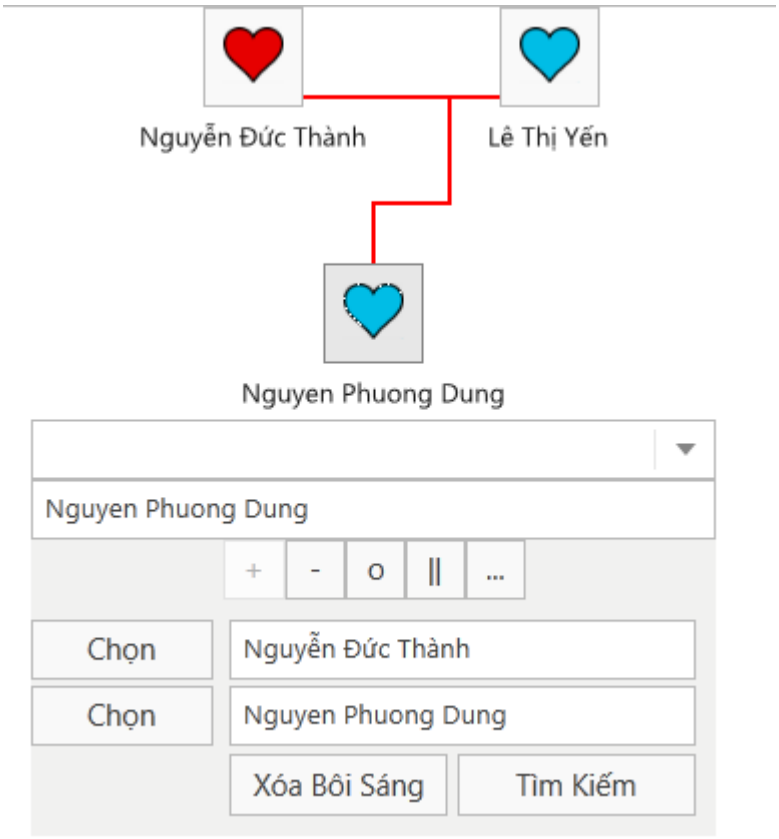
Step 2: Search: There are 2 options that you can choose to search relationship between 2 persons in a tree

Option 1:

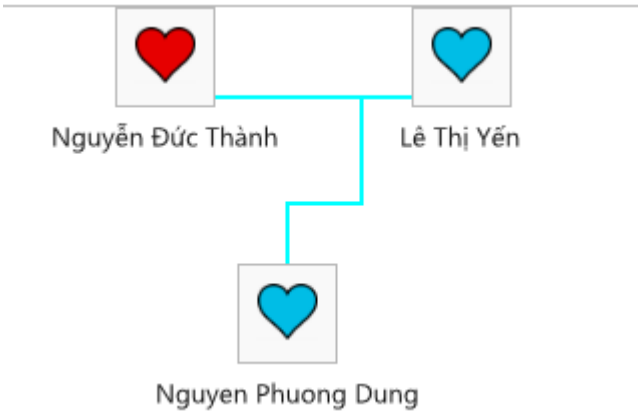
1.1 In the first node, click the "Chọn" to select the first person



1.2 Select a second node, and click 2nd "Chọn" button to select the second person



1.3 Click "Tìm kiếm" button ➔ the relationship between 2 persons will be highlighted

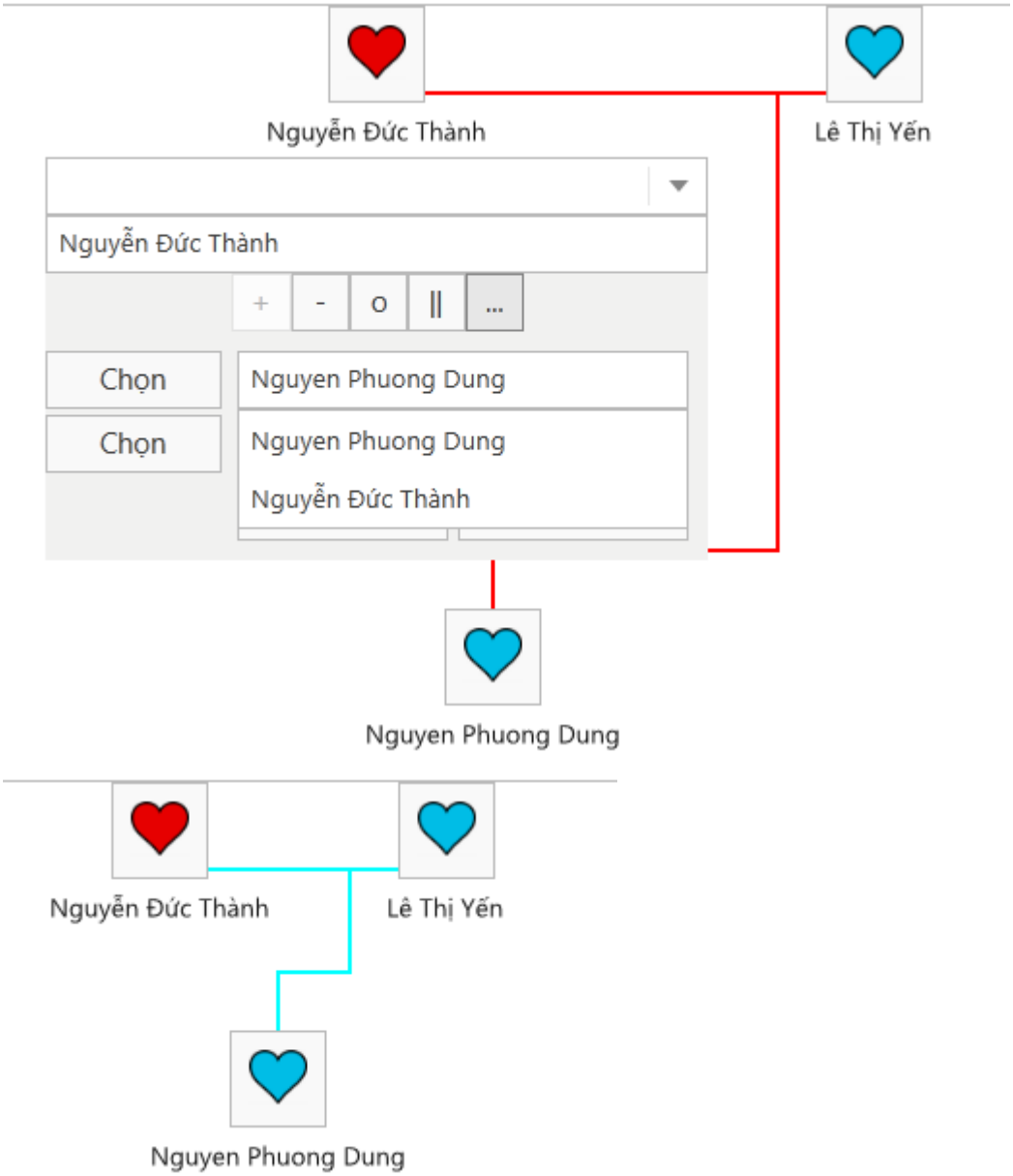


Option 2:

2.1 In the 1st "Chọn" text box, type name that you want to search

2.2 In the 2nd "Chọn" text box, type name that you want to search

2.3 Click "Tìm kiếm" button ➔ the relationship between 2 persons will be highlighted.



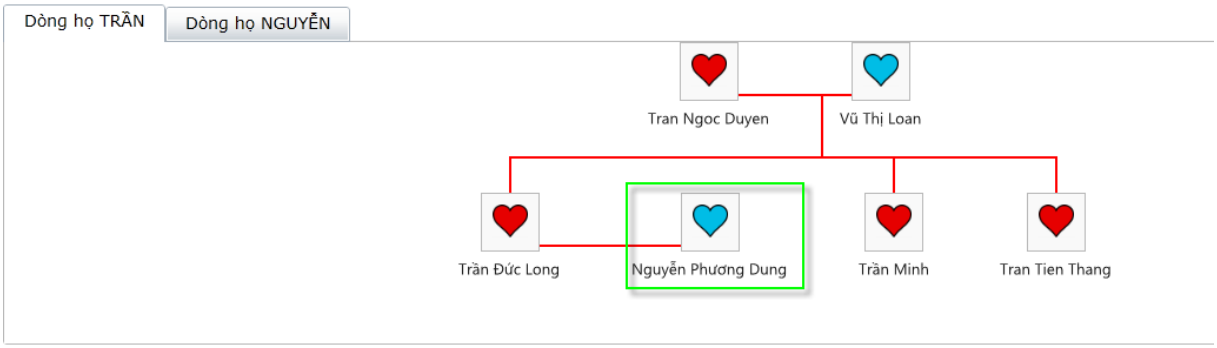
6.2.1.7 Search relationship between 2 persons in 2 trees

Pre-condition:

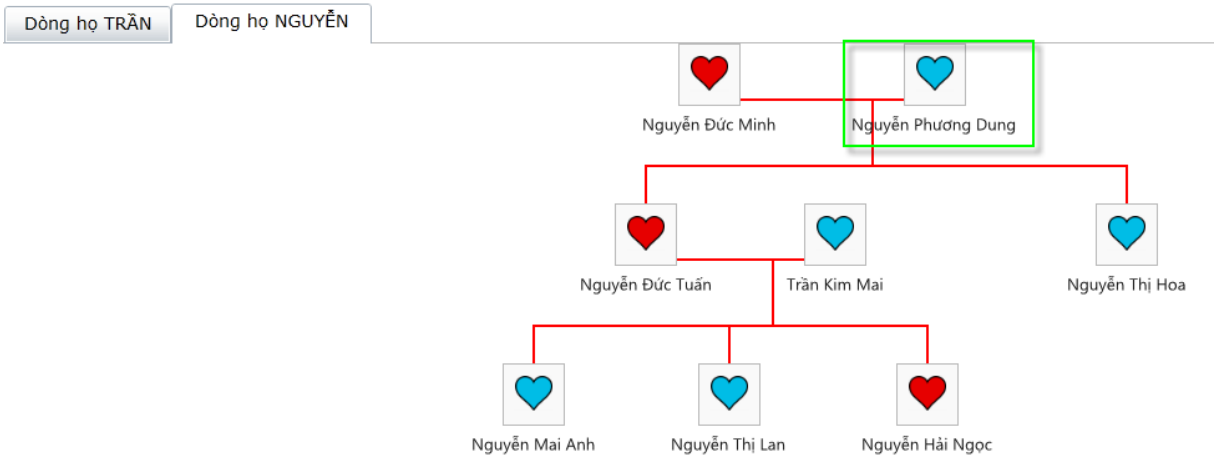
1. There are two trees in the “Cây gia phả” page

2. Between two trees, must be one node in common.

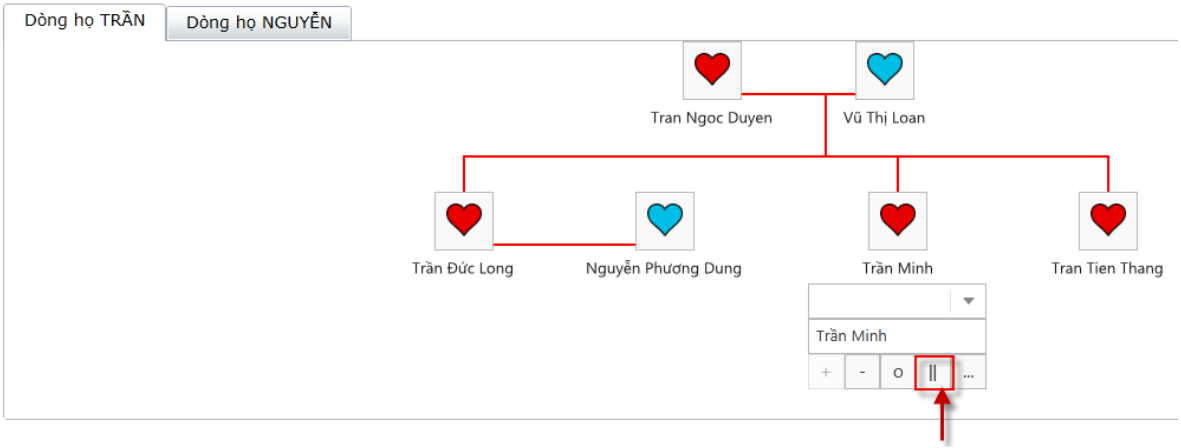
+ 1st Tree



+ 2nd



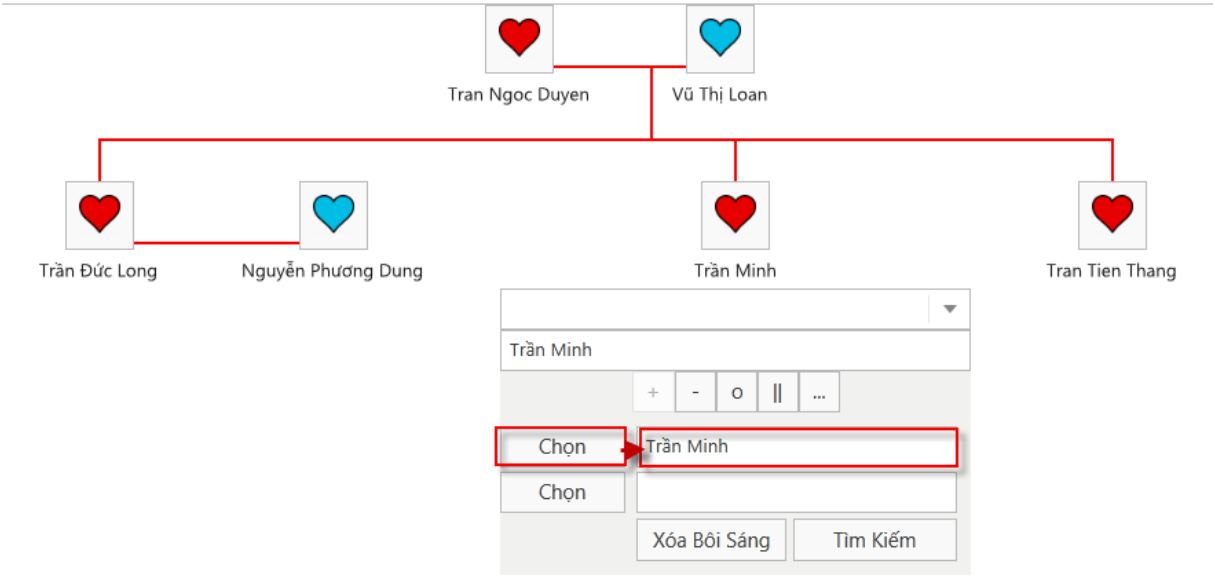
Step 1: In the 1st tree, select a node, click the "||" icon to display search screen



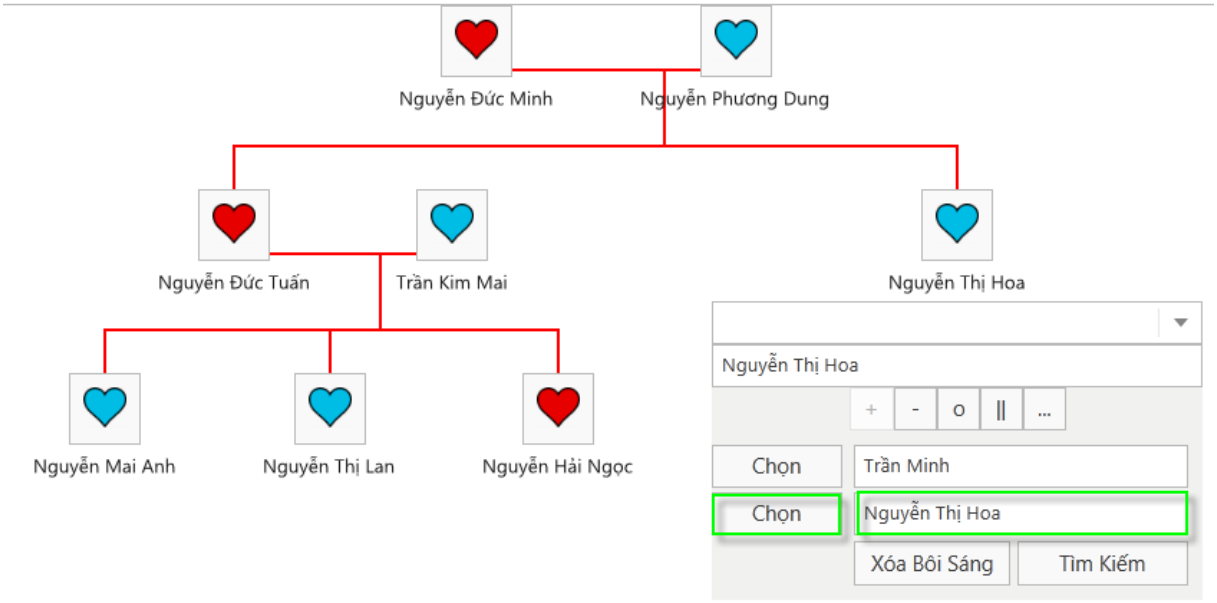
Step 2: Search: There are 2 options that you can choose to search relationship between 2 persons in a tree

Option 1:

1.1.In the 1st tree, select a node, then click the "Chọn" to select the first person

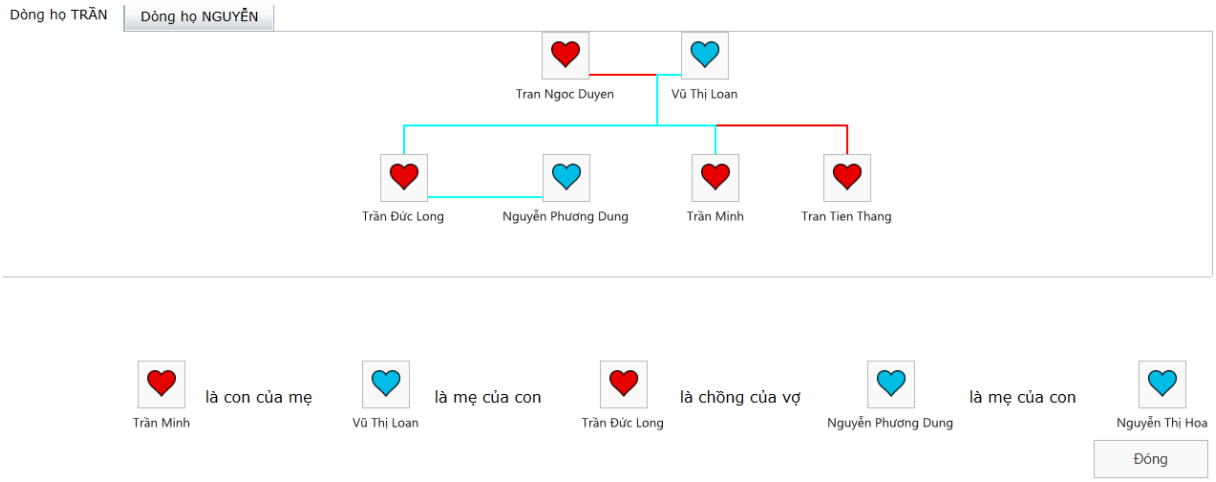


1.2.In the 2nd tree, select a second node, and click 2nd "Chọn" button to select the second person

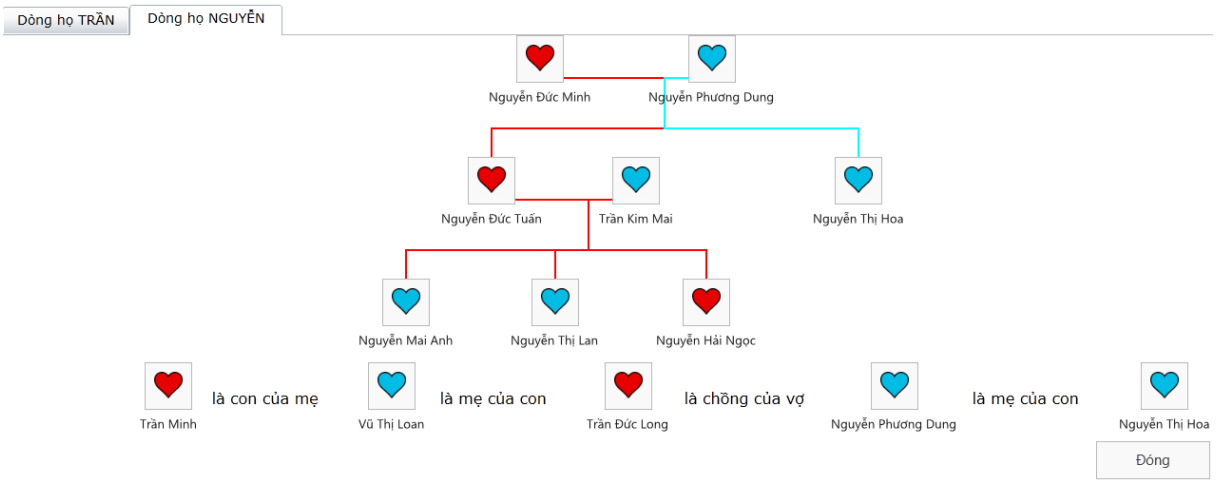


1.3.Click "Tìm kiếm" button ➔ the relationship between 2 persons will be highlighted

+ In the 1st tree

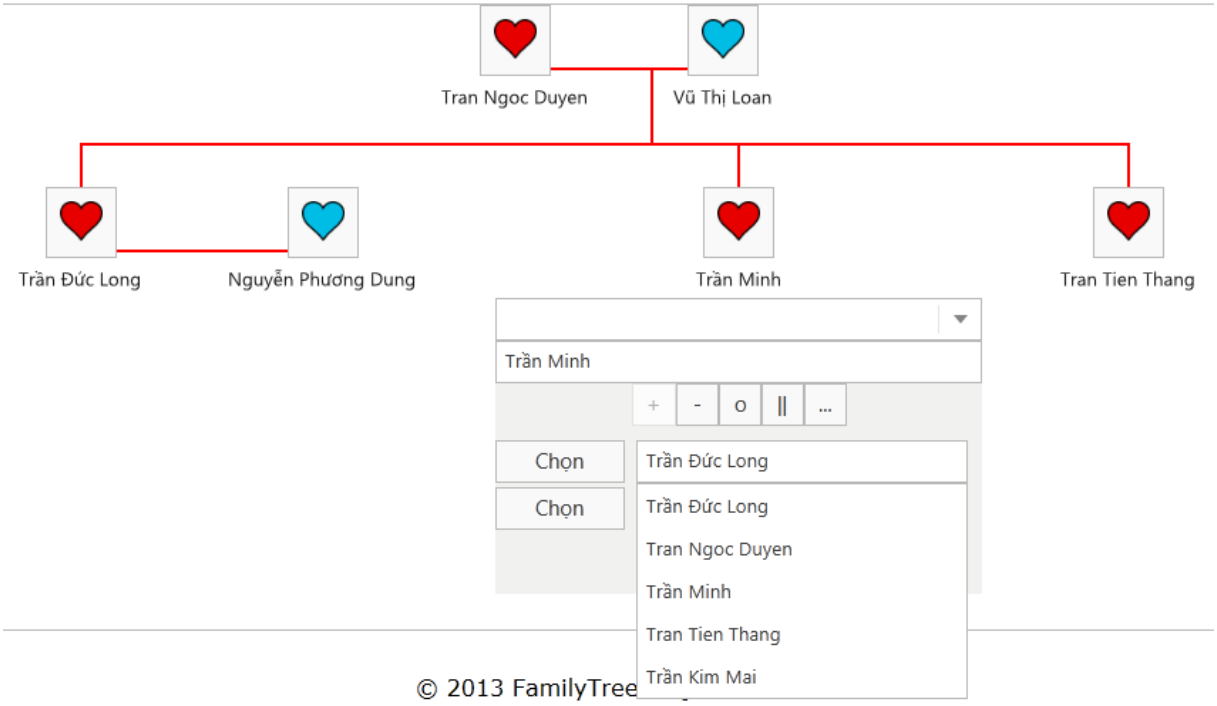


+ In the 2nd tree

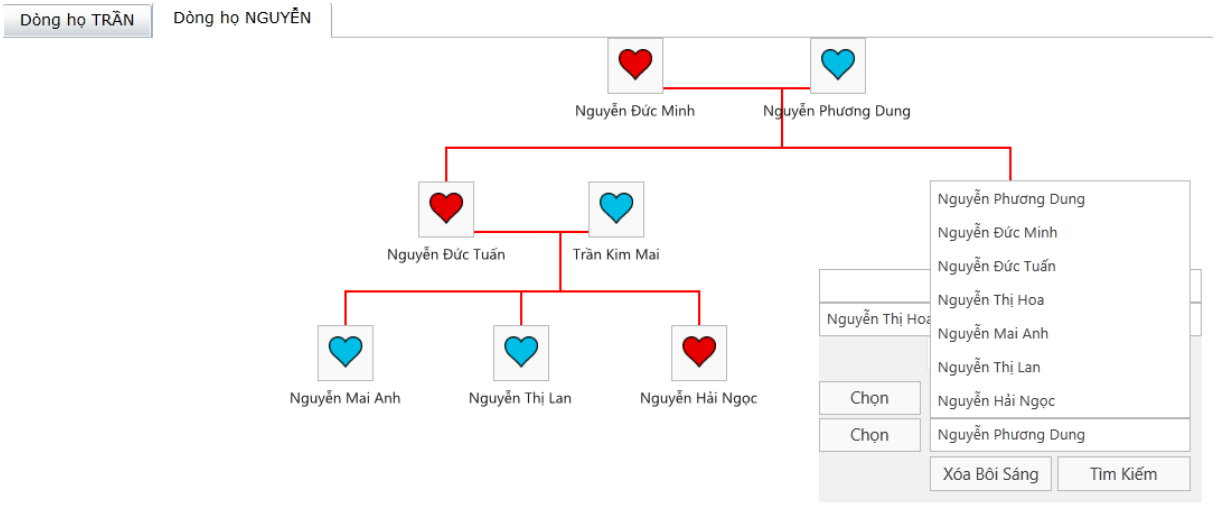


Option 2:

2.1 In the 1st "Chọn" text box, type name that you want to search

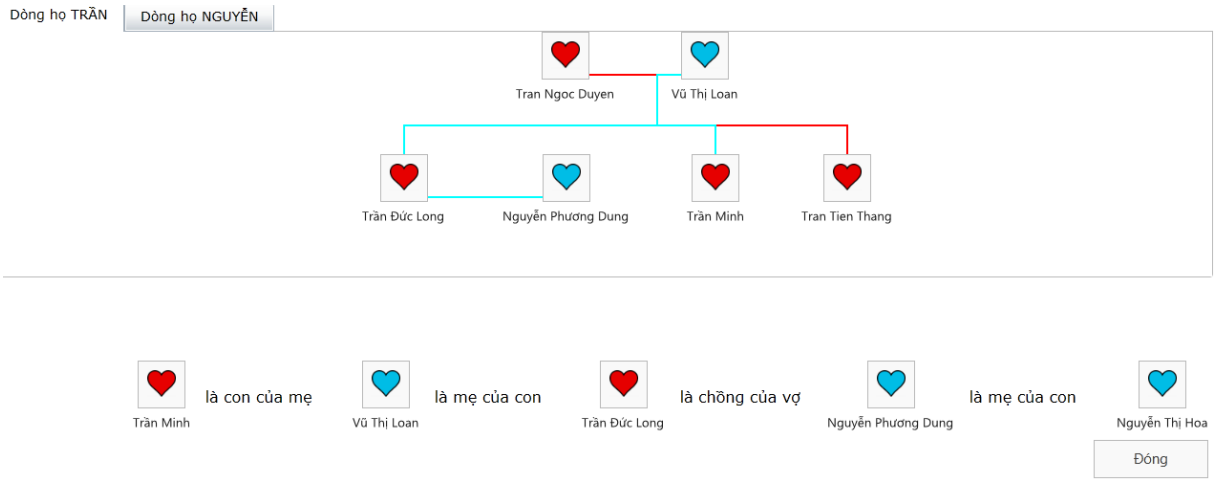


2.2 In the 2nd "Chọn" text box, type name that you want to search

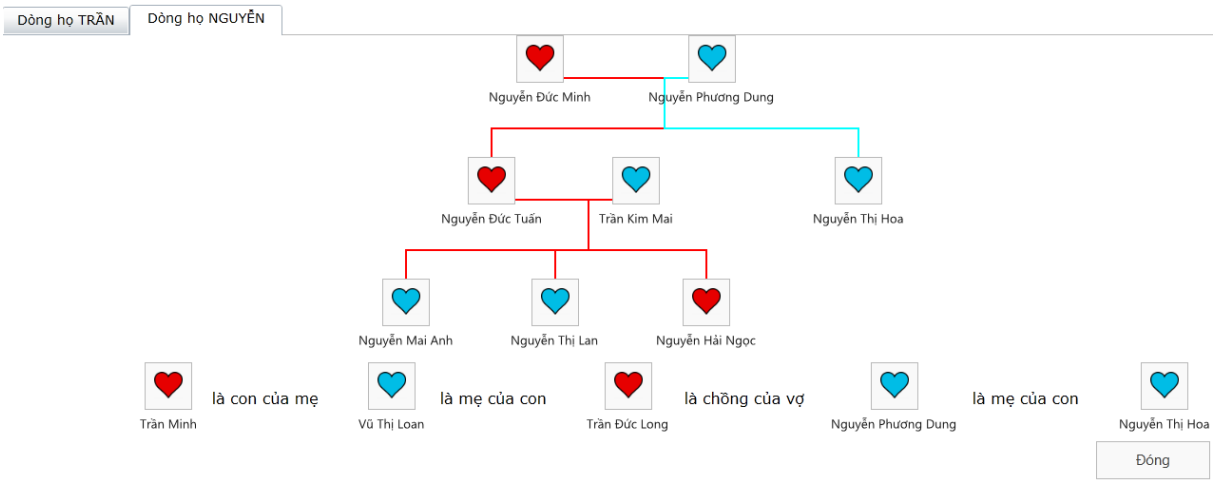


2.3 Click "Tìm kiếm" button ➔ the relationship between 2 persons will be highlighted.

+ In the 1st tree



+ In the 2nd tree

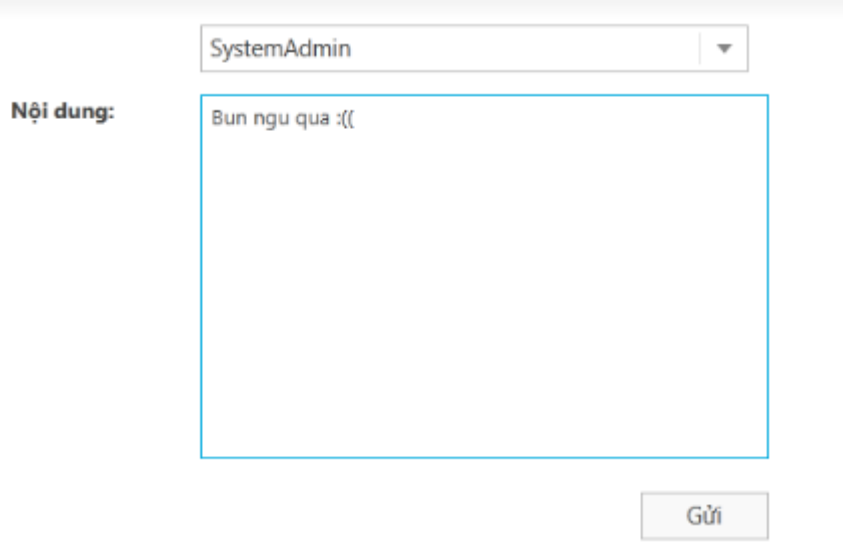


6.2.1.8 Send Request

4) From the main page, click the "Gửi yêu cầu" tab to send request
5) In the "Gửi yêu cầu" page, there are 2 sections:

+ "Người nhận": receiver
+ "Nội dung": content of request

6) After selecting receiver and typing the contect of request, you can click "Gửi" to send request or click "Hủy" to cancel send request.



6.2.1.9 Log Out

From the main page, click the "Đăng xuất" hyperlink in the top left corner to log out the Family Tree system

6.2.2. Family Administrator Guilde

1.2.2.1 Create account

Actions are same as Create Account actions from family user role.

1.2.2.2 Login

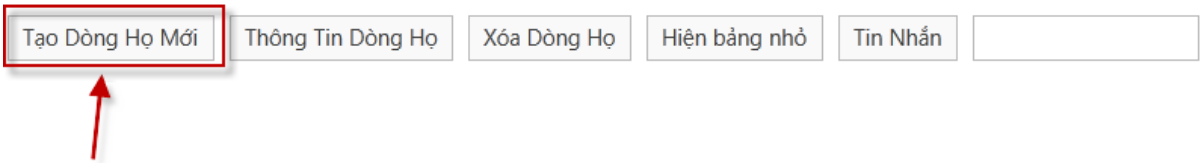
Actions are same as Login actions from family user role.

1.2.2.3 Default

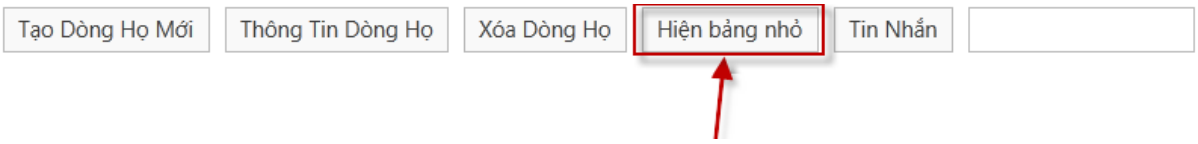
7) There are 6 items in the menu bar for the family administrator role:
    1. "Trang chủ": home page
    2. "Quản lý người dùng": list family user who join in tree
    3. "Cây gia phả": list of tree
    4. "Gửi yêu cầu": send request to system administrator screen
    5. "Liên hệ": some information about development team

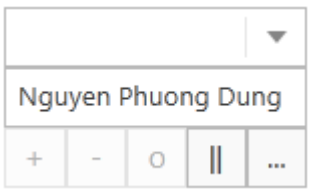
1.2.2.4 Create Tree

1. In the "Cây gia phả" page, click "Tạo dòng họ mới" button to add new a tree



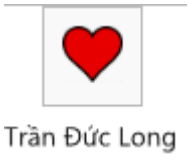
2. To add the first node, click "Hiện bảng nhỏ" button



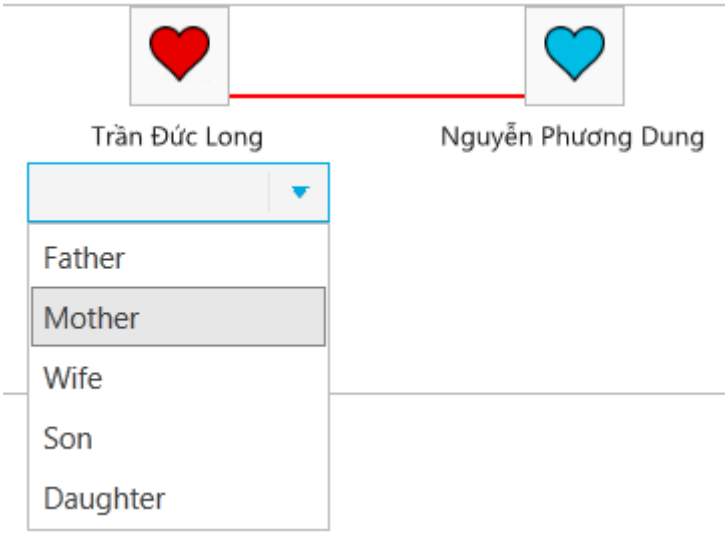


2.1."+" icon: add node
2.2."-" icon: delete node
2.3."o" icon: information of this node
2.4."||" icon: search relationship between 2 persons
2.5."…" icon: hide this table

3. Type name in the name text box, then click the "+" icon to add the first node

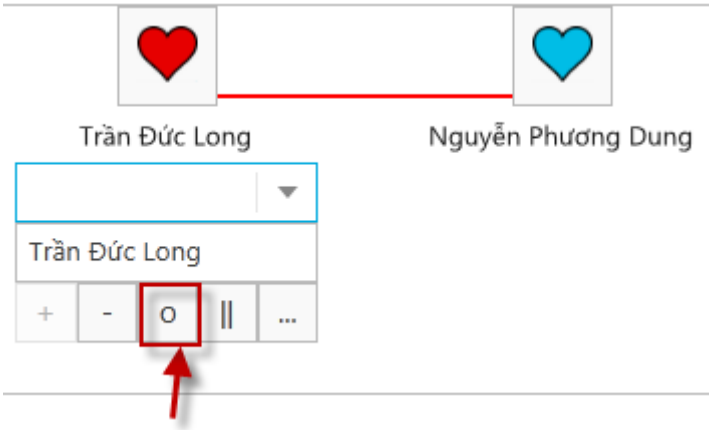


4. Add relationship for a person by select a node, and choose one of the relationship relationship as:Father, Mother, Wife (Husband), Son, Daughter

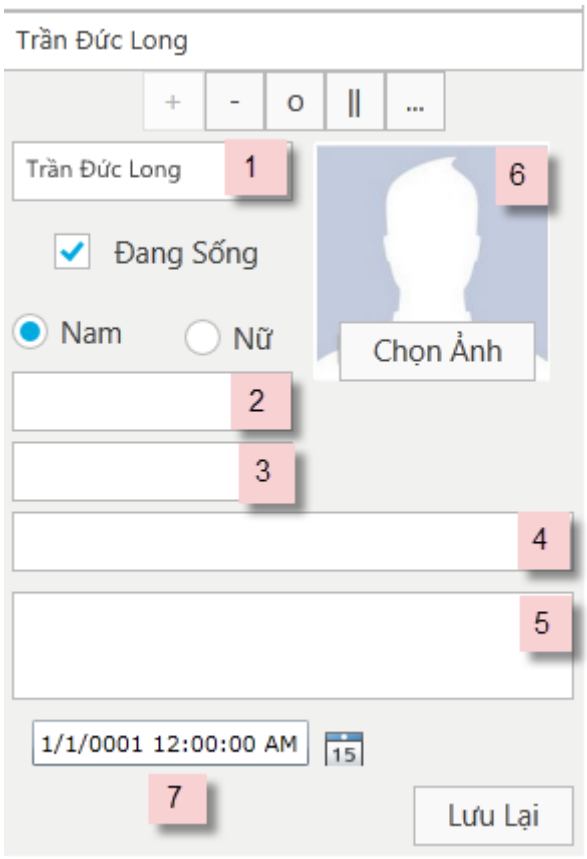


1.2.2.5 Edit node

1. Select a node ➔ display a small pop up as below:



2. Click the “o” icon to edit information ➔ Display edit information screen as below:

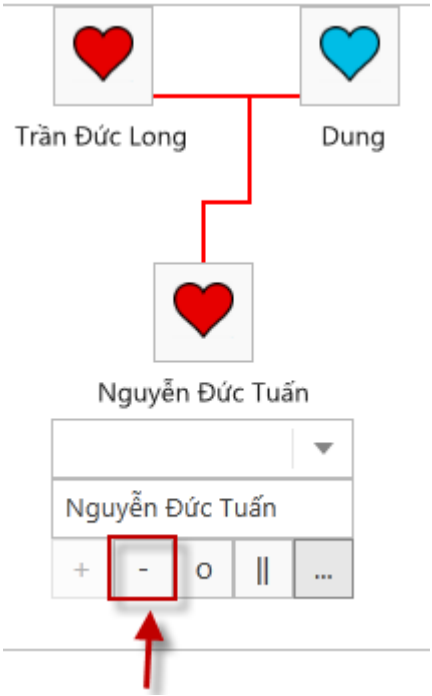


+ (1): Full name
+ (2): Profession
+ (3): What country are you in?
+ (4): Address
+ (5): Additional information
+ (6): Image of person
+ (7): Birth of date

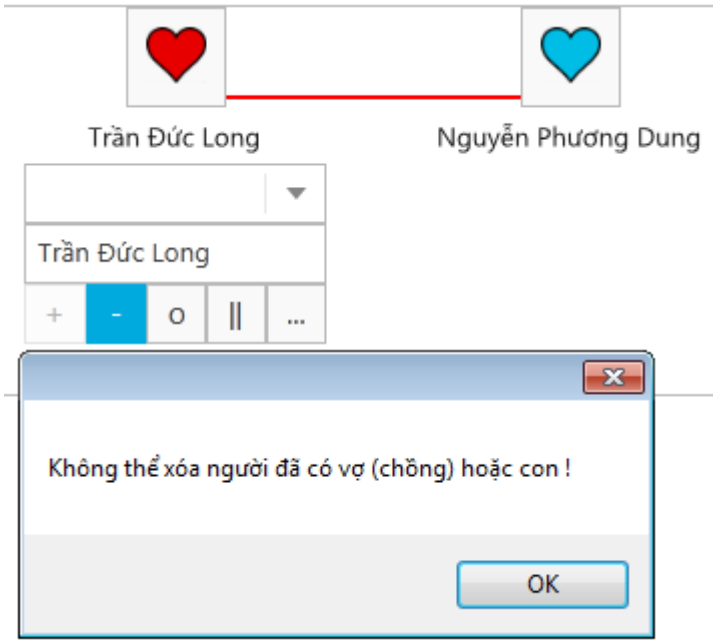
3. After editing information, click "Lưu lại" to save your changes.

1.2.2.6 Delete node

1. Select a node ➔ display a small pop up as below:

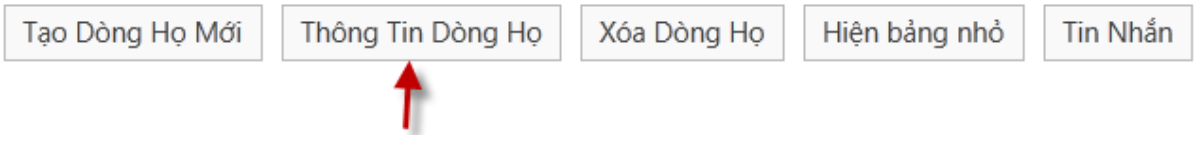


2. Click the "-" icon to delete a person
   Note that you cannot delete a person who has wife (husband) or son (daughter).



1.2.2.7 Edit tree's information

1. In the "Cây gia phả" page, click "Thông tin dòng họ" button to edit tree's information



2. Edit Tree's information screen is displayed as below:

3. Type tree information in:
   + (1): Name of family tree
   + (2): Name of head of family
   + (3): Family tree's address
   + (4): Additional information
4. Click "Lưu lại thông tin" button to save your changes or click "Hủy" to cancel your changes.

1.2.2.8 Search person

Actions are same as "Search person" actions from family user role.

1.2.2.9 Search relationship between 2 persons in a tree

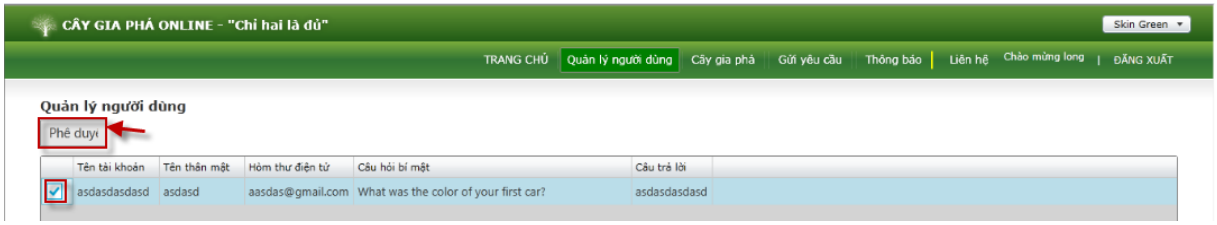
Actions are same as "Search relationship between 2 persons in a tree" actions from family user role.

1.2.2.10 Search relationship between 2 persons in 2 trees

Actions are same as "Search relationship between 2 persons in 2 trees" actions from family user role.

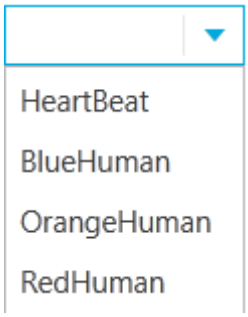
1.2.2.11 Approve family user

From the menu bar, click the "Quản lý user", and then navigate to "Quản lý user page"
This page contains list of family user who joins in family tree
Select a user, and then click on "Phê Duyệt" button to allow this user join in family tree



1.2.2.12 Customize tree

1. In the “Cây gia phả page”, click on the combo box in the top left corner to select one of the theme for tree



2. Select a theme, and tree will changes as your selected

2.1. HeartBeat type

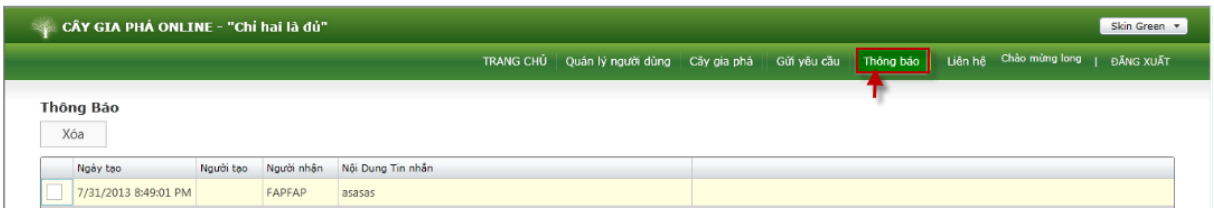
2.2. BlueHuman type

2.3. OrangeHuman type

2.4. RedHuman type

1.2.2.13 Review comment from family user

In the menu bar, click the “Thông báo” tab item to see the entire request from family user.



1.2.2.14 Log out

Actions are same as "Log out" actions from family user role.

6.2.3. System Adminnistrator Guide

6.2.3.1 Log in

Actions are same as "Log in" actions from family user role.

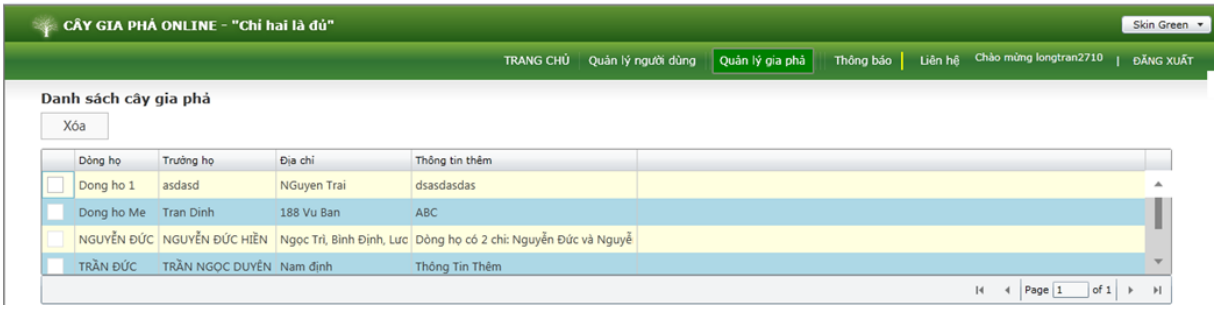
6.2.3.2 Default

There are 5 items in the menu bar for the family administrator role:

1. "Trang chủ": home page
2. "Quản lý người dùng": list uses who join in the system
3. "Quản lý gia phả": list of tree that have been created by family administrator
4. "Thông báo": List of request from family administrator
5. "Liên hệ": some information about development team

6.2.3.3 Delete tree

In the menu bar, click the "Quản lý gia phả" tab item to view list of tree in the system.

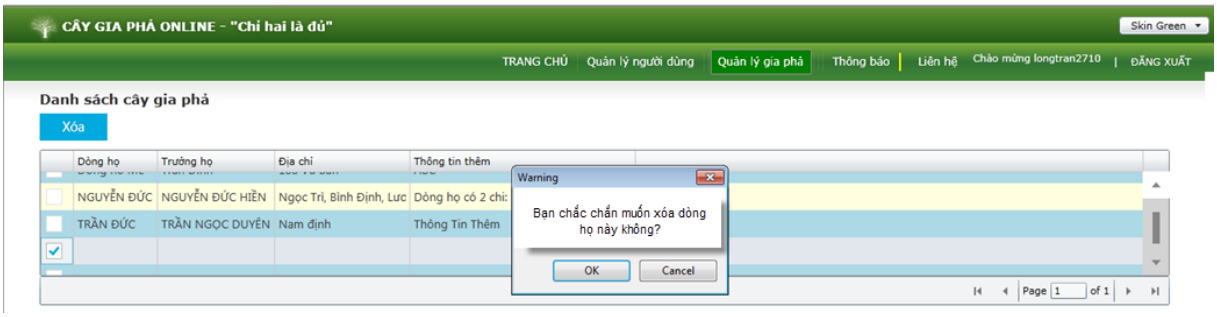


Then, you can select a tree by check in check box ➔ click the "Xóa" button

➔Display warning message as "Bạn chắc chắn muốn xóa dòng họ này không?" with 2 options: OK/Cancel

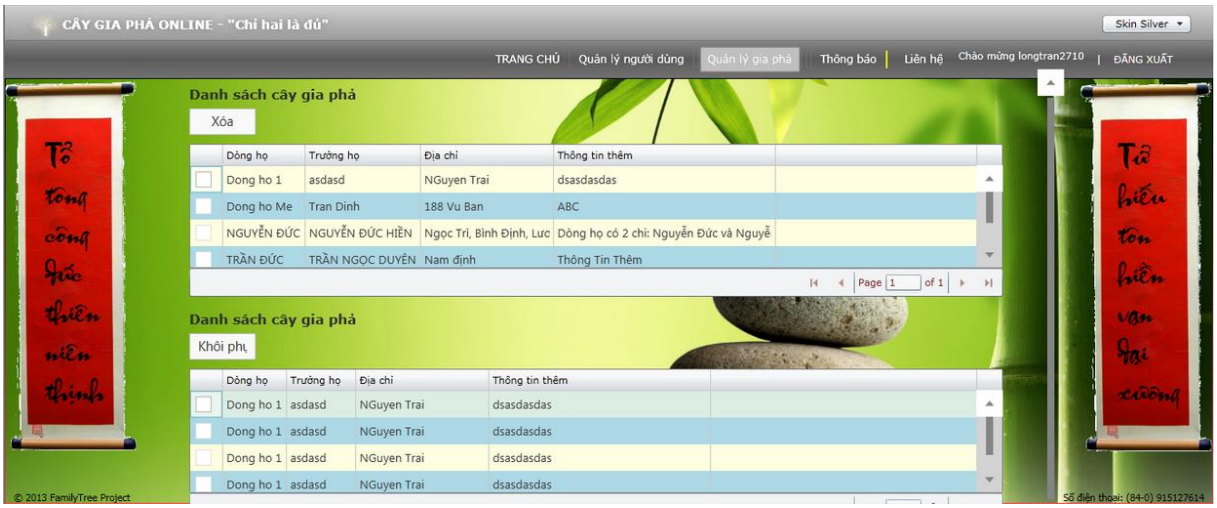
+ Click OK to confirm deletes the tree

+ Click Cancel to cancel delete the tree



6.2.3.4 Restore tree

In the menu bar, click the "Quản lý gia phả" tab item to view list of tree in the system.



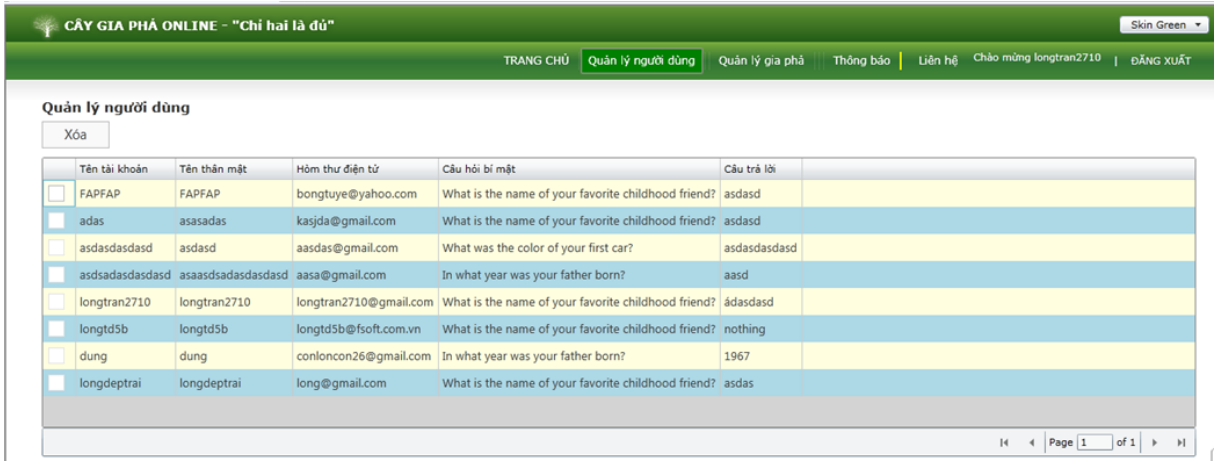
Then, you can select a tree by check in check box ➔ click the "Khôi phục" button

➔Display warning message as "Bạn chắc chắn muốn khôi phục dòng họ này không?" with 2 options: OK/Cancel

+ Click OK to confirm restores the tree

+ Click Cancel to cancel restore the tree

6.2.3.5 View user list



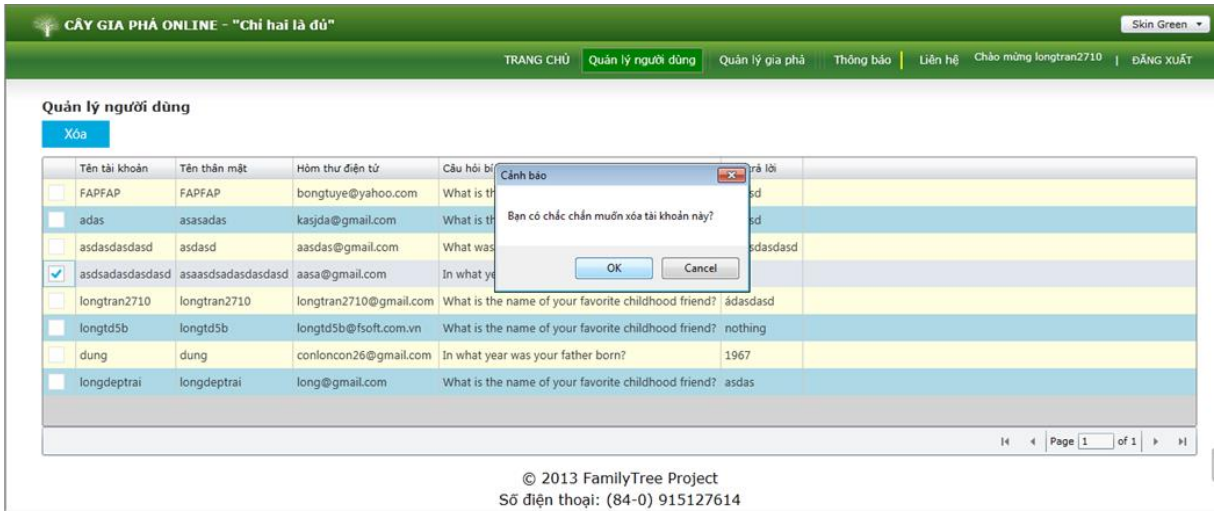
6.2.3.6 Delete user account

In the "Quản lý người dùng" page, select a person by check in the check box, then click the "Xóa" button

⇨ Display notice message as "Bạn có chắc chắn muốn xóa tài khoản này?" with 2 options: OK/Cancel

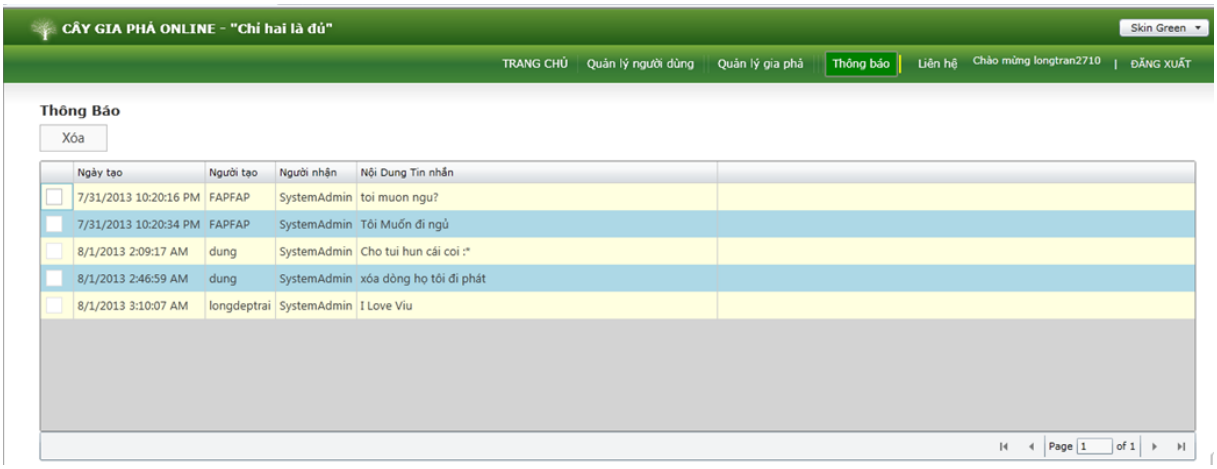
+ Click OK to confirm deletes this user

+ Click Cancel to cancel deletes user action



6.2.3.7 Review request from family administrator

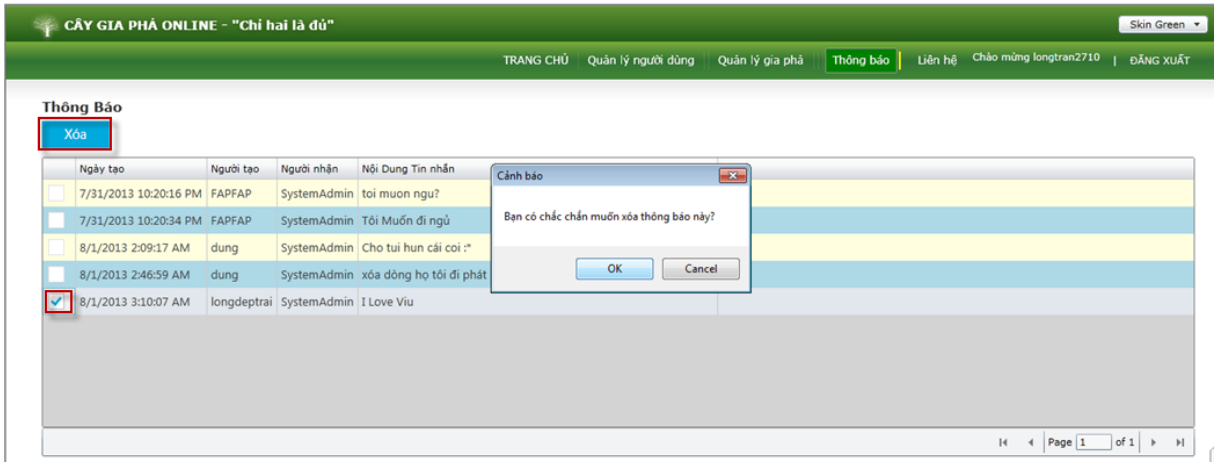
In the main page, click the “Thông báo” tab item to view the entire request from family administrator.



6.2.3.8 Delete request from family administrator

In the “Thông báo” page, select a request by check in the check box, then click the “Xóa” button

⇨ Display notice message “Bạn có chắc chắn muốn xóa thông báo này?” with two options: OK/Cancel
+ Click OK to confirm to delete request
+ Click Cancel to cancel to delete request



6.2.3.9 Log out

Actions are same as “Log out” actions from family user role.